

AFIT/GA/ENY/98M-01

**DAMAGE DETECTION USING PATTERN
CLASSIFIERS**

THESIS

**Eric D. Swenson, B.S.
Captain, USAF**

AFIT/GA/ENY/98M-01

Approved for public release; distribution unlimited

19980423 064

Disclaimer

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, the Department of Defense, or the United States Government.

AFIT/GA/ENY/98M-01

DAMAGE DETECTION USING PATTERN CLASSIFIERS

THESIS

Presented to the Faculty of the Graduate School of Engineering of the Air Force Institute of
Technology Air University In Partial Fulfillment for the Degree of
Master of Science in Astronautical Engineering

Eric D. Swenson, B.S.
Captain, USAF

Air Force Institute of Technology
Wright-Patterson AFB, Ohio
March 1998

Sponsored in part by Phillips Laboratory, Kirtland AFB, New Mexico

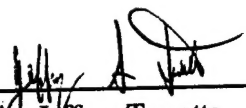
Approved for public release; distribution unlimited

DAMAGE DETECTION USING PATTERN CLASSIFIERS

Eric D. Swenson, B.S.


Captain, USAF

Approved:




Major Jeffrey Turcotte
Committee Chairman

5 MAR 98
Date



Dr. Bradley Liebst
Committee Member

6 MAR 98
Date



Dr. Martin Desimio
Committee Member

5 MAR 98
Date

Acknowledgments

I would like to express my sincere gratitude towards my advisors Major Jeffrey Turcotte and Dr. Bradley Liebst for their expert guidance, friendship and endless encouragement and support. I am grateful for all of the time they allowed me to ask questions, receive clear explanations, and get answers. I would also like to thank Dr. Martin Desimio and Captain Al Arb for their significant contributions in the area of classifiers and their technical review as without their help this thesis would not have been possible. I am, like others before me, indebted to Mr. Jay Anderson and his staff of technicians in the lab. Without their support and insight, options would have remained limited and avenues unexplored. Special thanks goes to my father Frank and my sister Donna for their help correcting errors and mistakes.

Most of all, I would like to thank my wife Kris and my sons Sean and Seth who were there for me through the whole AFIT experience.

Table of Contents

	Page
Acknowledgments	iv
Table of Contents	v
List of Figures	ix
List of Tables	xi
List of Symbols	xii
List of Abbreviations	xiv
Abstract	xv
Chapter 1. INTRODUCTION AND BACKGROUND	1
1.1 Introduction	1
1.2 Other Damage Detection Methodologies	3
1.3 Pattern Classifier Damage Detection Methodologies	3
1.4 Thesis Concept	6
1.5 Thesis Overview	8
Chapter 2. VIBRATION TESTING	9
2.1 Introduction	9
2.2 Flexible Truss Experiment	9
2.3 Experiment Setup	9
2.4 Data Collection	12

Chapter 3.	FINITE ELEMENT MODEL DEVELOPMENT	15
3.1	Introduction	15
3.2	Finite Elements	15
3.3	Previous Work on the Flexible Truss Experiment (FTE)	18
3.4	Development of Finite Element Models of FTE	21
3.4.1	Baseline Model	21
3.4.2	Stiff Model	27
Chapter 4.	MODEL UPDATING	31
4.1	Introduction	31
4.2	Selection of a Method Using FRF Data	31
4.3	Frequency Point Selection	32
4.4	Parameter Selection	33
4.5	Penalty Function	36
4.6	Optimization of the Penalty Function	36
4.7	Tuning Results	38
4.7.1	Tuning Results from Baseline FRF-Tuned Model	39
4.7.2	Tuning Results from Stiff FRF-Tuned Model	40
Chapter 5.	PATTERN CLASSIFIERS	47
5.1	Introduction	47
5.2	Data Variation	47

5.3	Feature Extraction	49
5.4	Pattern Classifiers	51
5.4.1	Gaussian Classifiers	51
5.4.2	Artificial Neural Networks (ANNs)	53
5.5	Damage Detection Results	55
5.6	Accuracy Rate Analysis	56
Chapter 6.	RESULTS AND CONCLUSIONS	60
6.1	Compilation of Results	60
6.1.1	Finite Element Model Development Results	60
6.1.2	Model Updating Results	60
6.1.3	Pattern Classifier Results	61
6.2	Research Conclusions and Recommendations	64
Appendix A.	FLEXIBLE TRUSS EXPERIMENT PROPERTIES AND CALCULATIONS	66
A.1	Flexible Truss Experiment	66
A.2	Material Properties	66
A.3	Longerons	67
A.4	Regular and Top Battens	67
A.5	Mid Battens	68
A.6	Diagonal Members	69
A.7	Vertical Plates	71

A.8	Horizontal Plates	72
A.9	Shakers	72
A.10	Additional Lumped Masses	73
A.11	Boundary Conditions	74
Appendix B. ELEMENT MATRICES CALCULATIONS		75
B.1	Stiffness Matrix	75
B.2	Mass Matrix	76
Bibliography		77
Vita		79

List of Figures

	Page
Figure 1. Block diagram for complete damage detection process	7
Figure 2. Diagram of experiment setup	10
Figure 3. Picture of experiment setup	12
Figure 4. Diagram of beam element with degrees of freedom	18
Figure 5. Baseline model of FTE	22
Figure 6. Truss section of baseline model	22
Figure 7. First bending, torsion, and breathing modes	24
Figure 8. Second bending, torsion, and breathing modes	24
Figure 9. Third bending, torsion, and breathing modes	25
Figure 10. FTE baseline model results	26
Figure 11. Diagram of rigid and base nodes	27
Figure 12. Diagram of nodes and stiff elements for vertical and horizontal plates	28
Figure 13. Diagram of the stiff model	29
Figure 14. Stiff model results	30
Figure 15. Tuning frequency points for nodes 33 and 36	33
Figure 16. FRFs from baseline model before tuning	42
Figure 17. FRFs from baseline after tuning using FRFs	43
Figure 18. FRFs from baseline after tuning using modal data	44

	Page
Figure 19. FRFs from stiff model before tuning	45
Figure 20. FRFs from stiff model after tuning	46
Figure 21. 100 FRF measurements overlayed with respective histograms of the variation in data	48
Figure 22. Block diagram of feature extraction process	49
Figure 23. Diagram of triangular filter bank	50
Figure 24. Example of measured test vector in 3-dimensional space	51
Figure 25. Diagram of a typical MLP	54
Figure 26. Damaged state showing measured and synthetic FRFs	59
Figure 27. Regular and top batten dimensions	67
Figure 28. Mid-batten dimensions	69
Figure 29. Picture of mid-batten	69
Figure 30. Picture of diagonal member	70
Figure 31. Diagram of diagonal cross section	70
Figure 32. Diagram of vertical plate	71
Figure 33. Picture of truss edge	71
Figure 34. Diagram of top plate	72
Figure 35. Picture of mid plate	72
Figure 36. Picture of shaker and accelerometers	73
Figure 37. Picture of the FTE Base	74

List of Tables

	Page
Table 1. Substructures for tuning the baseline model	34
Table 2. Substructures for tuning the stiff model	35
Table 3. Penalty function values before and after tuning	38
Table 4. Tuning parameter results from FRF tuning of the baseline model	39
Table 5. Results from FRF tuning of the stiff model	40
Table 6. Types of Gaussian classifiers	53
Table 7. Accuracy rate calculated from 300 synthetic training vectors from each class and 100 measured test vectors from each class results	56
Table 8. Error rate from By Class Diagonal	58
Table 9. Penalty function values before and after tuning	61
Table 10. Accuracy rate from 300 synthetic training vectors from each class and 100 measured test vectors from each class results	62
Table 11. Error rate from By Class Diagonal	63
Table 12. Aluminum properties	66
Table 13. Lexan properties	66

List of Symbols

Symbols	Definition (units)
$\square, \{\}$	Matrix, vector
$[b]$	Input shape matrix
B	Dynamic stiffness matrix (lb_f)
β_j	Tuning parameter
$[c]$	Sensor output shape matrix
C	Damping matrix ($\text{lb}_f\text{-sec/in}$)
E	Young's modulus of elasticity (lb_f/in^2)
ε_{OE}	Output error
$F(s)$	Applied force vector
γ, θ, ξ	Rotational degrees of freedom(radians)
$[H]$	Frequency Response Function
I	Second moment of area (in^4)
I, Γ, Ω^2	Modal mass (identity), damping, and stiffness matrices
J	Polar moment of area (in^4)
K	Stiffness matrix (lb_f/in)
L	Length (in)
λ_j	j^{th} Eigenvalue
M	Mass matrix (lb_m)
μ	Mean
N	Number of degrees of freedom (DOF)
NA	Number of shakers

NS	Number of sensors
ϕ_j	j^{th} Eigenvector
p	Modal coordinates
q	Model coordinates
R	Penalty Function
$s = j\omega$	Laplace variable
σ^2	Variance
$u(s)$	Input vector describing frequency input content of applied forces
u, v, w	Translational degrees of freedom(in)
ω	Frequency (radians/second)
$y(s)$	Output displacement vector
ζ_j	Non-dimensional modal damping factor

List of Abbreviations

Abbreviation	Definition
AFIT	Air Force Institute of Technology
ANN	Artificial Neural Network
APE	Assigned Partial Eigenstructure
ASTROS-ID	Automated Structural Optimization Software
DDE	Dynamic Data Exchange
DOF	Degrees of Freedom
FE	Finite Element
FEM	Finite Element Method
FRF	Frequency Response Function
FTE	Flexible Truss Experiment
MATLAB TM	Matrix Laboratory
MFCC	Mel-Frequency Cepstral Coefficients
MFSC	Mel-Frequency Spectral Coefficients
SA 390 TM	Scientific Atlanta Dynamic Signal Analyzer Model 390
SDT	Structural Dynamics Toolbox TM
WPAFB	Wright-Patterson Air Force Base

The research focused on developing and tuning finite element models to train pattern classifiers to detect and locate damage in a real structure. The research was broken into three distinct phases: finite element (FE) model development, FE model tuning, and pattern classifier training and testing. In the finite element development phase, a low order FE model called the baseline model and a high order model called the stiff model were created. In the FE model tuning phase, these FE models were tuned using measured Frequency Response Functions (FRFs), and the results were compared with previous research in which tuning was accomplished using modal data. In the pattern classifier training and testing phase, the tuned models were used to generate FRFs to train various pattern classifiers. Features (or properties) of the FRFs were extracted through an adapted feature extraction process commonly used in speech processing. This new feature set was developed to reduced the amount of data by a factor of 40 while retaining the salient properties that made the changes in the FRFs unique to each damage state.

The method was tested on the Flexible Truss Experiment (FTE) at the Air Force Institute of Technology (AFIT). The FE models were developed and tuned in the Structural Dynamics ToolboxTM for MATLABTM. To prove that the different features extracted from 32 damage states were unique, some initial tests were performed in which five classifiers were trained and tested using measured data. These tests resulted in no classification errors. Since the different damage states produced unique feature vectors, the majority of the research effort was spent developing and tuning different FE models that are then used to train five pattern classifiers to detect damage. Using the FE model generated FRFs, the Gaussian classifiers had an average accuracy rate of 78.25% for locating the correct damaged member and a 97% accuracy rate for locating the correct region of damage.

DAMAGE DETECTION USING PATTERN CLASSIFIERS

Chapter 1 - Introduction and Background

1.1 Introduction

In the last thirty years, major research efforts have been directed towards developing techniques for damage identification based on changes in the vibration characteristics of structures. Detection, location, and characterization of structural damage by measurement of the structural vibration response has been researched extensively. In 1969, Lifshitz and Rotem [13] published the first journal article that proposed the idea of detecting damage through measurable changes in vibration characteristics [5, 12]. Later, it was found that damage could be detected through changes in the vibration characteristics such as mode shapes, modal frequencies, displacements, or Frequency Response Functions (FRFs). Numerous damage detection routines have been applied to such structures as bridges, offshore oil platforms, planes, large buildings, etc. For further reference, a thorough overview of the damage detection field was completed in 1996 by Doebling, et al. [5].

As summarized by Natke [16, 2-9], a large structure can undergo many types of damage; e.g., cracks, holes, fractures, mass changes, material, or aging. These can be caused by: chemicals, accidents, unexpected forces, or defects. Time dependent effects such as aging, corrosion, or erosion may also cause changes in the vibration characteristics of a structure. Other effects like changes in the boundary conditions (e.g., a foundation crack) would also cause changes in characteristics. The capability to measure the changes in the vibration characteristics has been developed, but a need still exists to relate the changes to damage or to other causes.

A method to accurately and quickly detect damage in large structures has long been sought. The easiest method of detection is visual inspection, but this option is not always available or feasible (for example with space structures). The next easiest method of detection is local-measurement experiments, where experimental techniques such as ultrasound are used to find cracks. Again, this option may not be feasible or available. Expectations for a global damage detection methodology that can be applied to complex structures has led to the development of methods that examine changes in the vibration characteristics of a structure. The basis of these detection methods is that changes in physical parameters of the structure (mass, stiffness, or damping) cause changes in the structure's vibration characteristics.

Doebeling, et al. [5, 1-6] stated that an ideal method should detect damage, determine the location, and estimate the degree of damage. The method should be robust, that is insensitive to noise, changes in the structure's environment, and other non-linear and time-varying changes. Finally, an ideal routine should detect and classify any type of damage such as changes in stiffness, damping, or mass. To date, the methods that have been developed meet some of the ideal method's requirements. Conceptually, if there is a detectable and unique change in the FRFs, mode shapes, or modal frequencies, a method could detect these changes and classify a unique source of the changes.

This research investigated a damage detection method that would detect and locate damage caused by a loss of stiffness of a member in a large structure. This method used a pattern classifier to find the relationships between changes in the vibration characteristics and the damage that caused the changes. Although it does not meet all the requirements of an ideal method, it does show potential for being expanded to meet other requirements for an ideal damage detection method.

1.2 Other Damage Detection Methodologies

In 1969, Lifshitz and Rotem presented the first journal article on damage detection from changes in the natural frequencies of the structure [5, 12]. Because only a single measurement location is required, the changes in natural frequencies have turned out to be the most common and attractive indicator in all of the methods researched to date. Since 1969, there has been considerable research effort using mode shapes. However, to determine mode shapes, measurements must be taken at several points on the structure. Other methods such as matrix update methods, dynamic measured flexibility methods, non-linear, and mode shape curvature/strain mode shape changes have been developed over the years [5, 38]. From all of the damage detection methods that were available, pattern classifiers were chosen for this investigation because other researchers had achieved good results in damage detection on small test structures.

1.3 Pattern Classifier Damage Detection Methodologies

The section provides a quick summary of previous related research in date order and grouped by years. This section only covers research related to this thesis and is not comprehensive. For more background information see the referenced works or Doebling, et al. [5]. In the last eight years, pattern classifiers, such as artificial neural networks (ANNs), have been used to identify damage [5].

In 1992, Wu, *et al.* [23] used a multilayer perceptron (MLP) to identify damage in a simple three-story frame model driven by earthquake excitation. This MLP used 200 data points in a 0-to-20 Hz range from the Fast Fourier Transforms (FFTs) of acceleration measurements as the input. The MLP had one hidden layer with ten hidden nodes and three output nodes. Each output node represented the degree of damage in each floor. The network was able to diagnose the degree of damage within 25% on each of the first two stories.

In 1993, Elkordy, *et al.* [7] used a MLP to identify damage in five-story buildings. The damage was modeled as a reduction in stiffness in the bottom two stories, and the neural network was used to map changes in the first mode shape to correlate to changes in damage states. The network was trained on two 2-dimensional models of differing complexity and verified using measured data. This MLP was able to locate the damage and to determine the degree of damage in the first two floors to within 13% using their high order 2-dimensional finite element model.

In 1994, Kirkegaard and Rytter [10] used a MLP to detect damage in a 20-meter steel lattice mast subjected to wind excitation. Damage was modeled by varying the diagonal member's thickness. The network was trained from a finite element model using just the first five natural frequencies and tested using measured data. The researchers showed that a neural network trained with simulated data was capable of locating removed members of an actual structure. Manning [14] used a MLP on a 10-bar truss and a 25-bar transmission tower with active members. The imaginary parts of the pole and zero locations from the FRFs and an estimated measurement of member stiffness were used as the training data. The network was able to predict the change in member cross-sectional area to within 10%.

Povich and Lim [18] used a MLP to identify damage in a 20-bay planar truss composed of 60 struts. Damage was modeled by removing struts from the structure. Two accelerometers were used and FRFs were measured in a 0 to 50 Hz range, which included the first four bending modes. The ANN had 394 input data points and 60 outputs. The network was trained and tested using measured data only. The network was able to correctly identify 21 of the 60 members as damaged and 38 of 60 were narrowed down to 2 possible damaged members.

Stephens and Van Luchene [21] used a MLP to identify damage in multistory buildings. The training data came from a one-tenth scale model of a reinforced concrete structure. Damage was modeled by actually putting cracks in the concrete model. The network was trained on maximum

deformation, energy dissipation, and stiffness degradation as the structure responds to a seismic event using 60 data points. The network identified 25 of 32 test cases correctly. The investigators also tested a MLP on the Imperial County Services Building damaged in the Imperial Valley Earthquake of 1979 and correctly identified the building as lightly damaged.

Szewczk and Hajela [22] used a modified counterpropagational neural network as an inverse mapping between stiffness of individual elements global displacements under prescribed loads. All their tests were numerical simulations of 6, 9, and 14 element trusses. Using noise-free measurements, Szewczk and Hajela had no errors from their tests, and after noise was added with incomplete data, their error of predicting degree of damage stayed within 30%.

In 1995, Barai and Pandey [3] used a MLP to identify damage in a truss structure simulating a two-dimensional bridge. The authors used 70 time domain data points from a finite element model to train and test the neural net. Barai and Pandey report a prediction of the stiffness changes to within 4% accuracy in any member.

In 1996, Klenke and Paez [11] used probabilistic neural networks and a probabilistic pattern classifier to detect damage in an aerospace housing component. The neural network was used to determine whether the structure was damaged or not. If damaged, the classifier used the raw FRF data to determine the damage class. There were five different types of damage cuts made in the structure, and the classifier was able to clearly identify all five types of damage.

Doebeling, in [5, 49] said, "Identification of damage using neural networks is still in its infancy." Most researchers used MLPs as their pattern classifier. There was a variety of input sources to include: modal frequencies, mode shapes, FRFs, and static displacements. Most attempted to identify damage using modal frequency shifts. If only the modal frequencies are used, the chances of attributing unique changes to unique damage states are low. Therefore, the uniqueness of the changes in the vibration measurements are lost, and many damage states would look like they pro-

duced the same vibrational changes. On the other hand, researchers who used complete FRFs with several hundred data points found there was too much information for the pattern classifier to train on, resulting in long training times and large classifiers. Even though complete FRFs are much more robust and unique changes can be more easily measured, the amount of data can be overwhelming for a classifier. The key is to reduce the amount of data and still keep salient features so that unique damage states can be correctly classified.

1.4 Thesis Concept

The goal of this research was to develop and tune finite element models to train pattern classifiers to detect damage down to the element in a real structure. Several areas had to be explored to develop this complete damage detection algorithm. Figure 1 shows the entire process and which chapter in this thesis contains each part of the process.

The goal of the research was accomplished in three distinct phases: FE model development, FE model tuning, and classifier training and testing. To reach this goal, a good finite element model of the real structure had to be developed. Two models of the Flexible Truss Experiment (FTE) were developed and tuned. Development of the models started with measurement and calculation of the properties of the structure. This was followed by consideration of the boundary conditions, the level of detail (discretization), the types of elements, the choice of coordinates, and the number of input and output variables.

In the second phase, both models were tuned by changing stiffness and mass parameters in the model so as to closely match the measured FRFs using an optimization scheme to minimize the square of the log difference between measured and predicted FRFs. Tuning using FRF data directly was chosen over tuning using modal data because of the ease of use and reduction in error. The FRF tuned models were compared to a modal data tuned model developed in previous research on the

FTE [4]. The tuned models played the key role in the damage detection process. Thus, confidence in the results of the damage identification process were directly related to confidence in the tuned model.

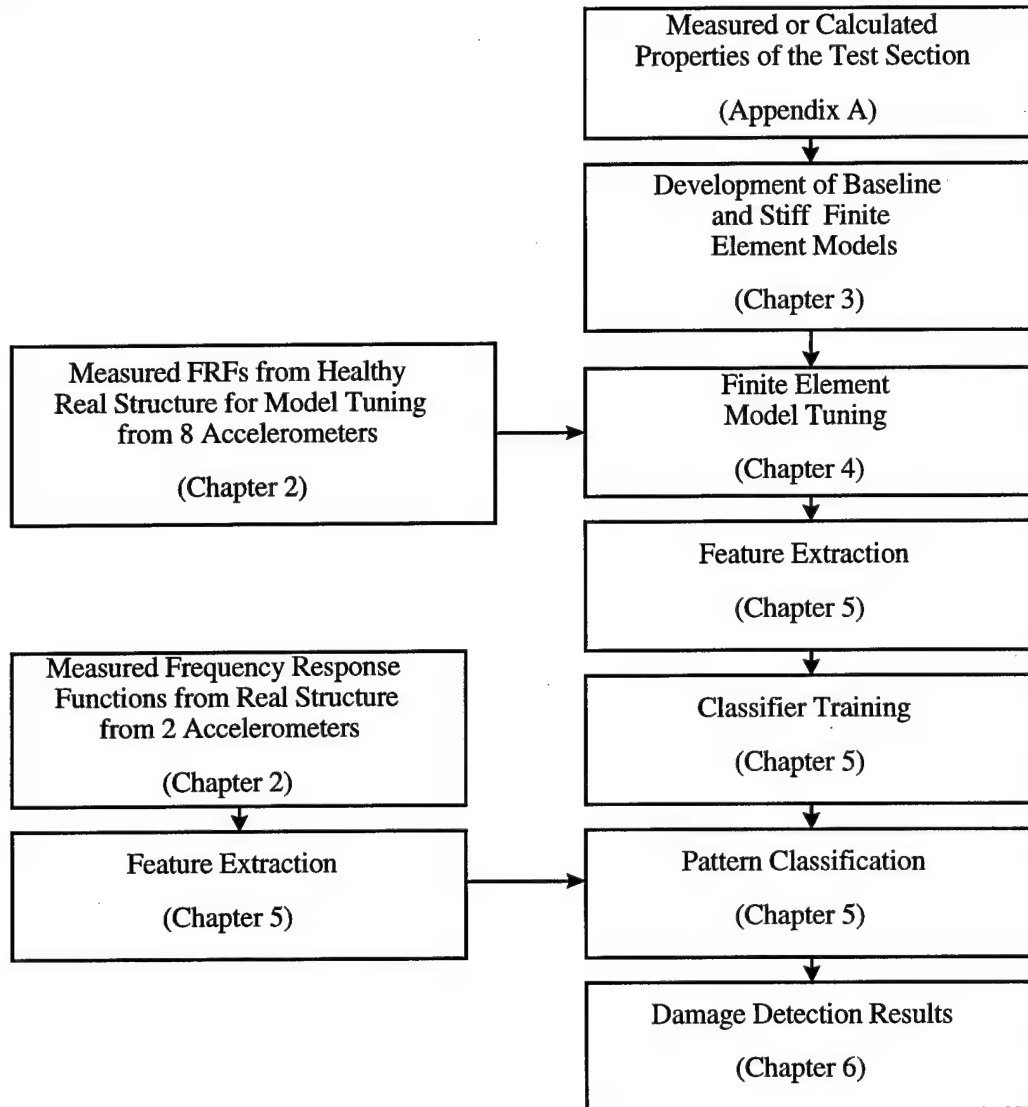


Figure 1. Block diagram for complete damage detection process

After model tuning was completed, the models were used to generate FRFs, referred to as synthetic FRFs, for all 32 damage states and the healthy state. The FTE is a 6 meter tall truss made of aluminum with bolt-in diagonal Lexan diagonal members. Each damage state was created by

reducing the stiffness in a diagonal member. These FRFs were then used to create a set of 100 FRFs by adding variation similar to the measured variation. These 100 FRFs for each damage state were then run through a feature extraction process. Various classifiers were trained and tested on the measured test data from the real structure to check their ability to locate/identify the damage.

1.5 Thesis Overview

The research contained in this thesis is organized in distinct development steps. In Chapter 2, the experiment setup is covered in detail. Chapter 3 includes the development of the finite element model and required modifications. Chapter 4 contains the tuning methods used to create a better model for training purposes. Chapter 5 describes the feature extraction process and how pattern classifiers work. Finally, in Chapter 6, the results of the experiment are summarized and conclusions are drawn.

Chapter 2 - Vibration Testing

2.1 Introduction

This chapter explains how the experiment was designed and developed and the data collected. It lists the decisions made on the type of data and how it was collected. The key instrument to success in this process was a good data acquisition and processing system. The Scientific Atlanta Pro Series Dynamic Signal Analyzer (SA 390) served as an excellent acquisition and analysis tool.

2.2 Flexible Truss Experiment

The truss sections of the FTE originally came from the 12-Meter Truss Active Vibration Control Experiment which was developed as part of the Wright Laboratories Large Space Structures Technology Program [9]. The FTE is a slender truss with a welded tubular aluminum alloy frame with a 20 square-inch cross section. The FTE is made from two 3-meter long sections with 4 bays per section. Each bay has four bolt-in Lexan diagonal members. The bays are welded together to create a back-to-back "K" pattern. The truss is bolted to a large aluminum plate at the bottom and free at the top. In Appendix A, the FTE is explained in detail along with the description of measurements and calculations of section properties. Figure 2 is a diagram of the experiment setup.

2.3 Experiment Setup

Two electromagnetic shakers were used at the top of the vertical truss to simultaneously excite the structure in the x and y direction (see Appendix A for more details). Both generated an input force approximately proportional to input voltage. By using two shakers, it was possible to excite both x and y translational modes, torsional modes, and breathing modes.

It was found that leaving both actuators on had no measurable shift in modal frequencies greater than $\pm .25$ Hz nor were there significant changes in amplitude. So, for this research, both actuators

were excited simultaneously with a common input voltage. By doing this, it was not necessary to take separate sets of x -excitation and y -excitation measurements.

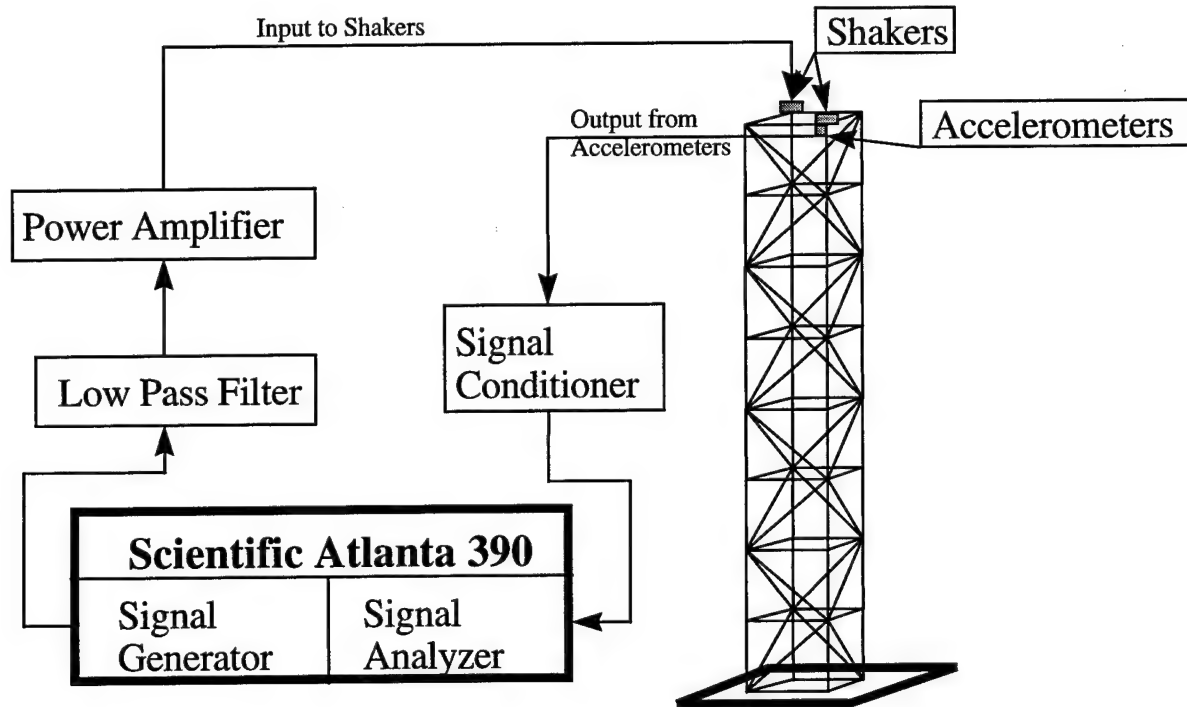


Figure 2. Diagram of experiment setup

Pseudo-random white noise was generated by the SA390 and input to the shakers. The input amplitude to the shakers was set to 2800 mvolts peak-to-peak. The generated noise was cutoff at 100 Hz by a low-pass filter and sent to the current drive circuits, which powered the shakers. Using an excitation hammer or externally-mounted shakers would not be feasible in space, and therefore are not considered for excitation. In this research, the shakers were bolted directly to the free end of the structure and were included as part of the model (as lumped masses). This caused only minor problems because these shakers only slightly changed the structure's dynamic characteristics [4].

The shakers were tested [9] and found to output a consistent force from 0-to-100 Hz except (due to a light centering spring) near 2 Hz, which was their natural frequency.

There were eight single-axis Sunstrand Q-Flex[®] accelerometers (see Appendix A for more details) bolted to the structure, four at the mid section and four at the free end. The accelerometers were aligned in the same orientation as the shakers and orthogonal to each other. Each accelerometer was included in the model as a lumped mass. After the signals from the accelerometers were amplified and conditioned, the signals were then digitized by the Pro Series Dynamic Signal Analyzer Scientific Atlanta 390 (SA 390).

The SA 390 is an eight-channel, 100 kHz analyzer that acquired, processed, and displayed the accelerometer data. The sampling rate was set to a low value of 256 Hz because the highest frequency sampled was only 100Hz with a 16-bit resolution. The sampling frequency needed to be at least twice the highest measured frequency to prevent aliasing. The problem of aliasing can be seen when two sinusoidal signals of different frequencies are measured as being at the same frequency because the sampling rate is too low. The solution to the problem was to have a low-pass filter with a sharp cut-off on the shaker signal and to have the sampling frequency at least twice the highest measured frequency.

After conversion from analog to digital form, the time domain data were converted to the frequency domain using the Fast Fourier Transform (FFT) algorithm. A major problem with digital vibrations data is leakage from one frequency to the next. To correct this problem, the input signal was multiplied by a windowing function called a Hanning window. The frequency content information was stored, and a FRF was computed by the SA 390 using the cross properties mode. An FRF is a frequency domain ratio of the response to the excitation force [20].

2.4 Data Collection

The SA 390 is a self-contained Intel 486 based computer with built-in digital signal processing boards. The analyzer ran the Microsoft[®] Windows[™] operating system which has the capability of multi-tasking. Typically, the Pro Series Analyzer acquisition software and MATLAB[™] were run concurrently. The SA 390 unit is shown in the bottom right corner of Figure 3.

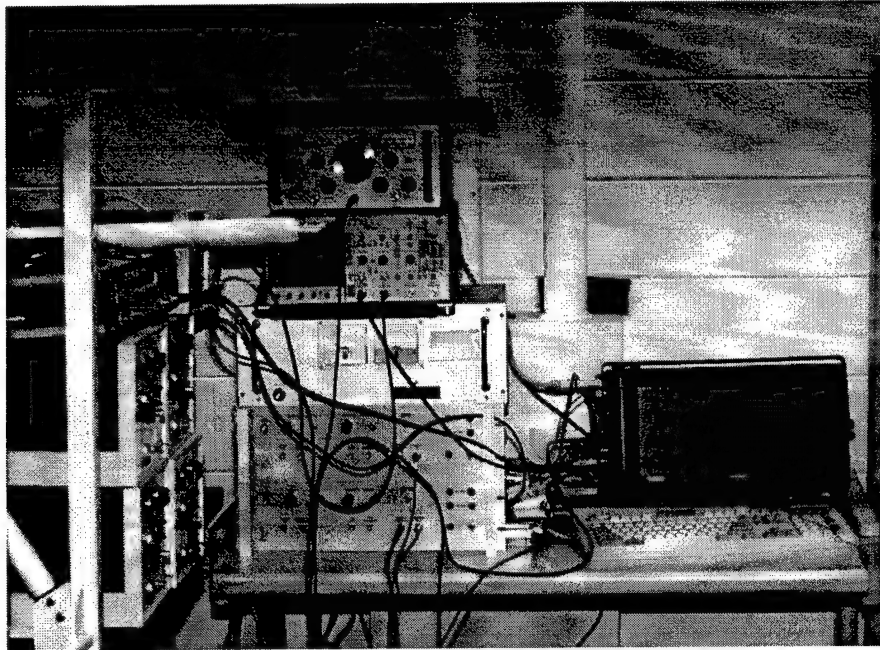


Figure 3. Picture of experiment setup

A complete FRF from 0-to-100 Hz with 400 data points was generated from 1,024 time domain samples. This process took almost four seconds. Another FRF was computed and averaged in with the first using a process called overlapping. The overlapping process was used to significantly reduce the time required to compute 100 averages. Overlapping allows 36 new data points to be added and 36 of the oldest data points to be dropped from the previous sample for the next FRF computation. This resulted in 100 averages being computed in 18 seconds.

Even though the SA 390 had a powerful acquisition program, it was limited in its capability to measure and compute 100 averages multiple times. After computing the 100 averages, the program had to be manually reset to take another sample. Fortunately, the software could be controlled through the Dynamic Data Exchange (DDE). The DDE allows data or commands to be sent between two WindowsTM programs running at the same time. A program was written in MATLABTM to control the SA 390 through the DDE. A MATLABTM m-file in the computer told the SA 390 when and what to do. This program allowed thousands of averaged FRFs to be measured without the continuous attention of the operator.

In the first part of the experiment, FRFs needed to be collected for tuning the finite element model. All eight accelerometers were used, and the number of averages was set to 2,000. This data set was referred to as the FRF tuning data. All of the plots in this thesis that compare the measured data versus predicted or synthetic responses used this data. Data from the eight accelerometers were used create a better model in the tuning process. By using more accelerometers, the tuning algorithm had more points to tune on thus creating a more physically correct model.

In the second part of the experiment, 6,600 FRFs were computed which had 100 averages per FRF taking a total of 16.5 hours of sampling time. Only two accelerometers on the top of the truss were used under the assumption that in a real system only limited data would be available for damage detection. The Flexible Truss Experiment (FTE) had 32 diagonal members that could be disconnected to simulate damage. By disconnecting one end of the diagonal member, a complete break or loss in rigidity could be simulated. For this experiment, there were 32 damage states and 1 healthy or undamaged state for a total of 33 states or classes of damage.

There were 100 FRFs measured from accelerometers under and in the same direction as the shakers. Each FRF had to have variability in the data so each class could be defined. By keeping the number of averages low, the measured FRFs had enough variability without conflicting classes. The

variability of the data is covered in detail in Chapter 5. All of these measurements resulted in 6,600 FRFs which were then converted to Cepstral Coefficients as explained in Chapter 5. This complete set of data was referred to as the measured data set. This data set was used in the experiment for training and testing the pattern classifiers.

In the last part of the experiment, only two FRFs were measured from accelerometers at one of the top nodes for testing the pattern classifier. Even though the first set of 6,600 FRFs were primarily used for testing the pattern classifiers, single measurements needed to be taken to demonstrate the capability of damage detection methodology to tour groups that came through AFIT. Several demonstrations were given where someone picked a member at random and damage was simulated by unbolting the member. The MATLABTM m-file had the SA 390 collect 100 averages, and then the trained pattern classifier was used to classify the damage.

Chapter 3 - Finite Element Model Development

3.1 Introduction

A short development of the equations used in this investigation is covered in this chapter to allow the reader to better understand the model updating process in the later chapters. The finite element method (FEM) creates a system of ordinary differential equations in matrix form to represent the mass and stiffness of a continuous structure. Fully developed mathematical formulations for the FEM can be found in any finite element textbook. The two most important characteristics of a finite element model are that it be physically significant and correct [1]. Physical significance means the model correctly represents the distribution of mass, stiffness, and damping. Correctness refers to the correlation between the dynamic response of the finite element model and the structure. In this chapter, the Structural Dynamics ToolboxTM for MATLABTM was used to create a finite element model of the Flexible Truss Experiment (FTE) that was physically significant. The models were tuned to make them more correct by adjusting parameters in the mass, stiffness, and damping matrices. The tuning process is discussed in Chapter 4.

3.2 Finite Elements

The goal of developing FE models of the FTE was to have them behave as closely as possible to real structure in all damage states. This section was written using the Structural Dynamics Toolbox manual [2] as a reference, for more details see this manual. The finite element representation used for this continuous structure is a set of second-order, coupled differential equations written in the Laplace domain.

$$\begin{aligned} [Ms^2 + Cs + K]_{N \times N} \{q\}_{N \times 1} &= [b]_{N \times NA} \{u(s)\}_{NA \times 1} \\ \{y(s)\}_{NS \times 1} &= [c]_{NS \times N} \{q\}_{N \times 1} \end{aligned} \quad (1)$$

This equation set establishes a relationship between the model coordinates $\{q\}_{N \times 1}$ and the output displacement $\{y(s)\}_{NS \times 1}$. The M , C , and K respectively represent the mass, damping, and stiffness matrices. These three matrices are $N \times N$, where N is the number of degrees of freedom (DOFs). NA is the number of actuators or shakers, and NS is the number of sensors. The applied forces can be written as $\{F(s)\} = [b] \{u(s)\}$ where the input shape matrix $[b]_{N \times NA}$ represents the location, direction, and scaling information from each input and the input vector $\{u(s)\}_{NA \times 1}$ defines the frequency properties of the input. The output shape matrix $[c]_{NS \times N}$ contains the location, direction, and scaling information from each output. The following block diagram shows the clear input/output relationship.

$$\{u(s)\} \xrightarrow{[b]} \{F(s)\} \xrightarrow{[Ms^2 + Cs + K]^{-1}} \{q(s)\} \xrightarrow{[c]} \{y(s)\}$$

The response vector $y(s)$ of the model in equation (1) can be decomposed as the sum of independent oscillators (modal responses). If the damping matrix C is small, it can be ignored and the response of the model can be decomposed into normal modes. Normal modes are determined from the eigenvector equation

$$-[M]_{N \times N} \{\phi_j\}_{N \times 1} \omega_j^2 + [K]_{N \times N} \{\phi_j\}_{N \times 1} = \{0\}_{N \times 1}$$

where, for a positive definite mass and positive semi-definite stiffness matrix, there are N independent eigenvectors ϕ_j and N eigenvalues ω_j^2 which is the square of the undamped natural frequency for each mode. The eigenvectors ϕ_j represent the modeshapes and the eigenvalues ω_j^2 represent the square of the undamped natural frequencies. There were many eigensolving techniques available, some of which are suited to take advantage of the structure of this problem. In finding the undamped natural frequencies, the Structural Dynamic Toolbox (SDT) gave a choice to use either the Lanczos algorithm or subspace iteration technique to compute the eigenvalues and eigenvectors from sparse matrices. Since the eigenvectors are orthogonal to the mass and stiffness matrices, equation (1) can

be transformed into the following form

$$\begin{aligned} [Is^2 + \Gamma s + \Omega^2] \{p\} &= [\phi^T b] \{u\} \\ \{y\} &= [c\phi] \{p\} \end{aligned}$$

where $\Gamma = \phi^T C \phi$ is the modal damping matrix and $p = \phi q$ is the modal coordinates. Ω^2 is the diagonal matrix of eigenvalues, ϕ is the mass normalized matrix of eigenvectors, $c\phi$ is the modal output shape matrix, and $\phi^T b$ is the modal input shape matrix.

The modal damping matrix Γ is generally not diagonal but will be assumed so in order to simplify the tuning process. The non-zero (diagonal) terms of the modal damping matrix can be expressed as $\Gamma_{jj} = 2\zeta_j \omega_j$, where the ζ_j are the modal fractions of critical damping. Such a model is said to have proportional or modal damping. The proportional damping approximation is commonly known as the simplest damping model next to no damping [8]. At this time, there is no direct physical justification for selecting a proportional damping model, but it did work for this lightly-damped system. The proportional damping model did cause considerable trouble in tuning the FE models because the damping parameters were adjusted for the accelerometers at the top of the truss, but the heights of the peaks were off on the accelerometers in the middle of the truss.

The technique used to determine approximate damping ratios was to compare the undamped FRF to the measured responses. The heights of the undamped responses were then adjusted to match the measured data by adjusting the damping ratios [8]. The SDT package also allowed a damping coefficient to be added to each mode. This made for more accurate matching between the measured and synthetic FRFs.

The standard Bernoulli-Euler 12 degree of freedom (DOF) element based on linear interpolations for traction and torsion and cubic interpolations for flexion was used in the model for all sections of the FTE. The element DOFs are the translations and rotations of each node, as seen in Figure 4. The element K and M matrices came from the section properties of each element. The

construction of each element matrix is shown in Appendix B. The properties of the three types of elements used in this investigation, battens, longerons, and diagonals are covered in Appendix A. The element K and M matrices were assembled to create global K and M matrices, and each matrix was $N \times N$. The mass matrix was a consistent mass matrix, where the total mass of the beam was distributed among the DOFs. See Appendix B for more details.

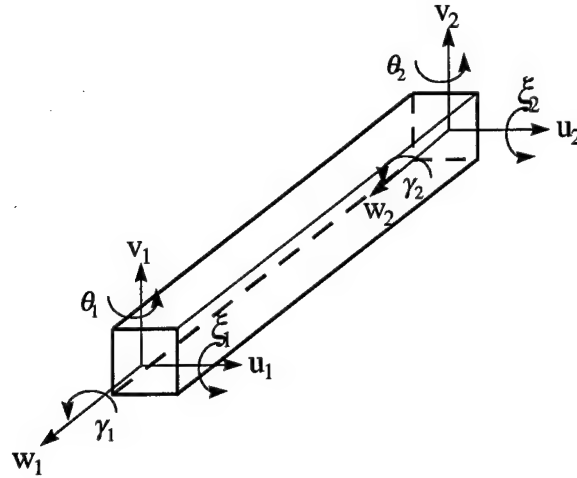


Figure 4. Diagram of beam element with degrees of freedom

3.3 Previous Work on the Flexible Truss Experiment (FTE)

Over the years, several finite element models have been created to model the FTE in some form or another. The FTE was developed as part of the 12-Meter Truss Active Control Experiment conducted at the Flight Dynamics Directorate of Wright Laboratory [9]. Two finite element models were created with varying degrees of success in the experiment. After the experiment was completed, the original truss was taken apart and given to AFIT. Presently, only two of the original four sections are joined together to create the 6-Meter FTE.

In the 12-Meter Truss Active Vibration Control Experiment two FE models were created to predict open-loop response. A frame model very similar to the baseline model in this thesis and a

continuous beam model were developed. The original researchers did not get their frame model to agree with measured values. Their model predicted torsion and bending mode frequencies below the measured frequencies. Attempts were made to manually adjust the elastic modulus of the members to get the predicted and measured responses to match. The researchers stated "All of the parameters had significant effects on the truss frequencies, but it was not clear what combination of adjustments was optimal and physically plausible."

The researchers gave up on their frame model and developed a continuous beam model where each bay was modelled as a single beam element. Again, they had trouble with torsion and bending modes. They decreased the elastic modulus by 30% and increased the torsional stiffness by 21% to force the model to agree with the measured results. They concluded "The physical explanation of why such large changes in bending and torsion parameters were not clear." After reorganization of the Wright Laboratory in the early 90's, the truss sections were given to AFIT, and two of the sections were installed in the AFIT labs and renamed the Flexible Truss Experiment (FTE). Since then, research on the FTE has changed focus from active control to testing damage detection methodologies.

Captain Richard Cobb, a previous PhD student at AFIT, completed his dissertation *Structural Damage Identification Using Partial Eigenstructure* [4] using the FTE. He focused his research on developing a new method to identify structural damage using limited modal data. His experiment included creating a model of the FTE, tuning it using modal data in a package called Automated Structural Optimization Software adapted for model tuning (ASTROS-ID), and identifying damage using a newly developed assigned partial eigenstructure (APE) method. Cobb used the first eight modal frequencies and their corresponding modeshapes. His research showed that damage could be localized to a small section of the truss.

Cobb created his FE model in NASTRAN, and it had 100 elements, 36 nodes, and 216 degrees of freedom. The baseline model developed in this thesis came from Cobb's baseline model with a few modifications. Cobb also presented a sensor prioritization method that was used to prioritize the location of the eight accelerometers for damage identification. The prioritized locations were at nodes: 36, 33, 20, and 17, as shown in Figure 5 on page 22.

After determining the best sensor locations and actuator positions, Cobb extracted the mode shapes and modal frequencies from the FTE. Cobb had trouble extracting some of the mode shapes because they were too close together. This high modal density resulted in modeshapes being measured that are linear combinations of other shapes.

After extracting the mode shapes, he was able to use this modal data to tune his baseline model using ASTROS-ID. Using ASTROS-ID, Cobb tuned his baseline model to match the measured modeshapes. He had 23 tuning parameters: I_1 , I_2 , E , mass for each of the 5 member types (except mass on the top battens), and 4 lumped mass parameters: vertical plates, top plates, mid plates, and actuators. The resulting parameter values ranged from 0.42 to 2.81, and the tuning process took 18 minutes of CPU time on a SPARC 20. For this thesis, Cobb's baseline and baseline modal-tuned models were re-created in the SDT for MATLABTM. The tuned model gave good results and was used as a basis of comparison between using modal data or FRFs to tune the baseline model. The results of the tuning comparison can be found in Chapter 4, and the results of using the different tuned models were compared again in Chapter 5 when used for damage detection. Cobb's tuned model was referred to as the baseline modal-tuned model.

After tuning the baseline model using the modal data, Cobb created a damage localization method to determine the detectability of certain damage states. Cobb found 26 batten elements were undetectable and the remaining 70 elements fell into 15 symmetric groups. His APE method was then used to determine which group the damage came from. He tested 2 damage states and

was able to correctly identify the small area of the truss containing the true damaged element. The calculations took on average of 9 minutes on a SPARC 20. Cobb concluded, "The extent to which damage can be localized was limited by both model fidelity and accuracy of the measured modes."

3.4 Development of Finite Element Models of FTE

The FE model, shown in Figure 5, was created in MATLABTM using the Structural Dynamics ToolboxTM (SDT). This toolbox, created by Etienne Balmes [2], was designed as a low-cost and easy-to-use set of tools for creating high-fidelity finite element models of structures. The toolbox includes tools for experimental modal analysis, FE analysis, and model updating [2]. The remainder of this chapter covers the baseline model of the FTE and then covers the high-order stiff model of the FTE.

3.4.1 Baseline Model

An initial finite element model, called the baseline model, was created in MATLABTM using verified data from a pre-existing NASTRAN model developed by Richard Cobb [4]. The model had 36 nodes, 96 elements, and 216 degrees of freedom. The material properties and section properties were carefully remeasured or rederived from the FTE and verified against the pre-existing model (see Appendix A for development of the values). Most of the measured or derived quantities came from previous work on the FTE [4, 9]. There were many simplifications made when modeling the FTE. These led to accepted errors and inconsistencies in the model. Joints and boundary conditions were the most difficult parts to model correctly. The first model was called the baseline model. The joint conditions were simplified and then more precisely modeled. The FRFs generated by the FE models is referred to as synthetic data.

Unfortunately, the FRFs of the baseline model at nodes 33 and 36 did not match up with the measured data (see Figure 10 on page 26). The heights of the peaks were generally dependent on

the damping in the model, but the locations of the peaks were dependent on the mass and stiffness properties of the structure. Individual peaks were affected by adjusting various parts of the model. Shown in Figures 7, 8, and 9 is a wire-frame diagram of the structure deforming under the different modes. The scale factor of the deformation was increased to exaggerate the deformation of the structure. One can see how certain members were affected by certain modes. For example, the first bending modes were very dependent on axial stiffness in the longerons.

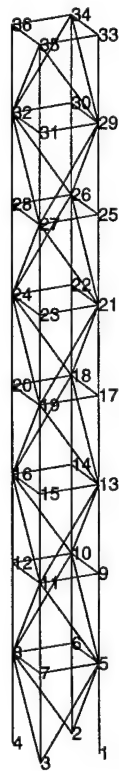


Figure 5. Baseline model of FTE

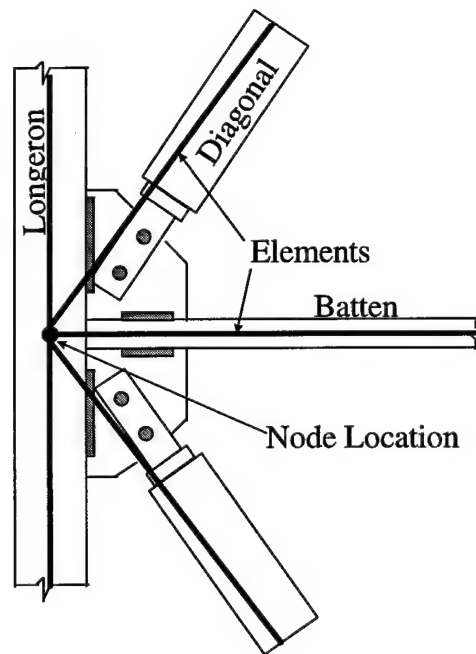


Figure 6. Truss section of baseline model

After careful study and experimentation, the following conclusions were drawn about the baseline model. The breathing modes were studied and found to be mostly dependent on the stiffness in the battens and their joints. Since the first, second, and third breathing modes were lower in frequency than they should have been, there was a component of stiffness missing or too much mass in the battens.

The torsion modes were studied and found to be more dependent on the stiffness in the diagonals, rigidity in the joints, and mass at the nodes. In the baseline model, the first, second, and third torsion modes were lower in frequency than they should have been. After careful experimentation, it was found the torsion peaks were dependent on the location of the lumped masses. In the baseline model, all of the lumped masses were located at the nodes which placed the center of mass further from the centerline of the truss. To test how much this influenced the torsion modes, the lumped masses were moved closer to the center of the structure, thus reducing the structure's total rotational inertia. This reduction in inertia increased the frequencies of the torsion modes and made the model more realistic, because the mass of the diagonal end sockets is actually located closer to the center of the structure.

After further research, the causes of the mismatch between the breathing and torsion modes were determined to be the absence of welded plates throughout the model and the lumping of all the mass at the node points. The original baseline model had all intersecting members connected at single nodes. The effects of the plates and the bolted connections were not modeled correctly. Stiff elements were added to the baseline model to include the stiffness effects of the welded plates and bolted connections. Additionally, the masses added by the plates and diagonal member ends were displaced away from the original node. The details of these modifications are covered in the next section.

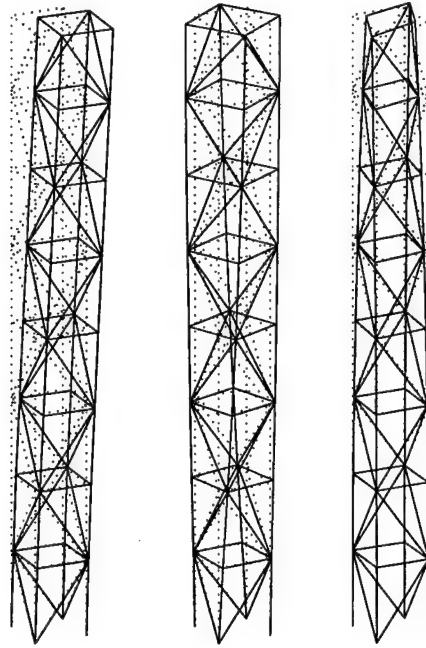


Figure 7. First bending, torsion, and breathing modes

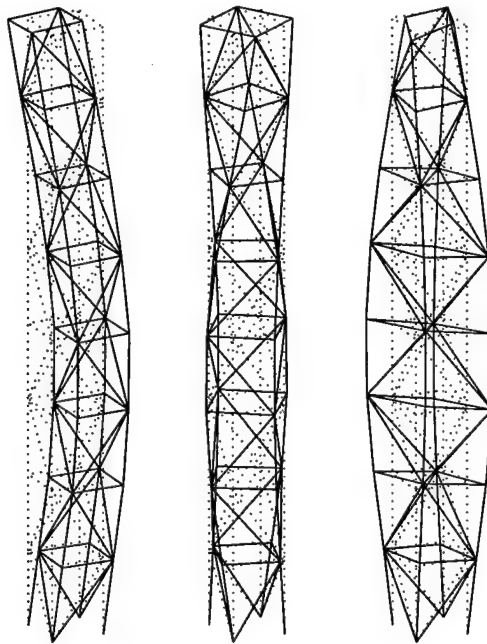


Figure 8. Second bending, torsion, and breathing modes

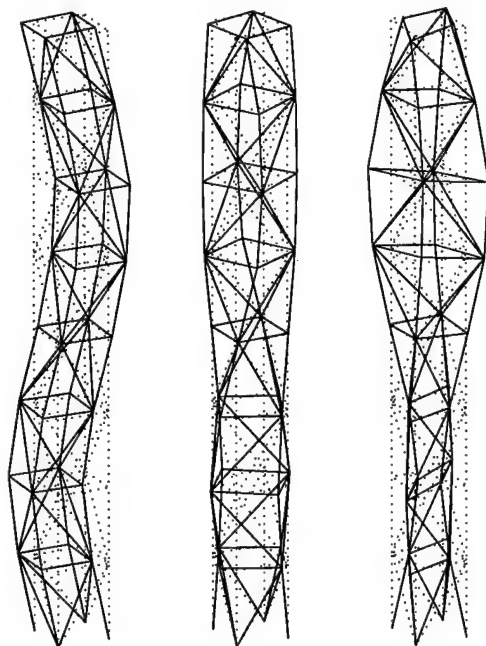


Figure 9. Third bending, torsion, and breathing modes

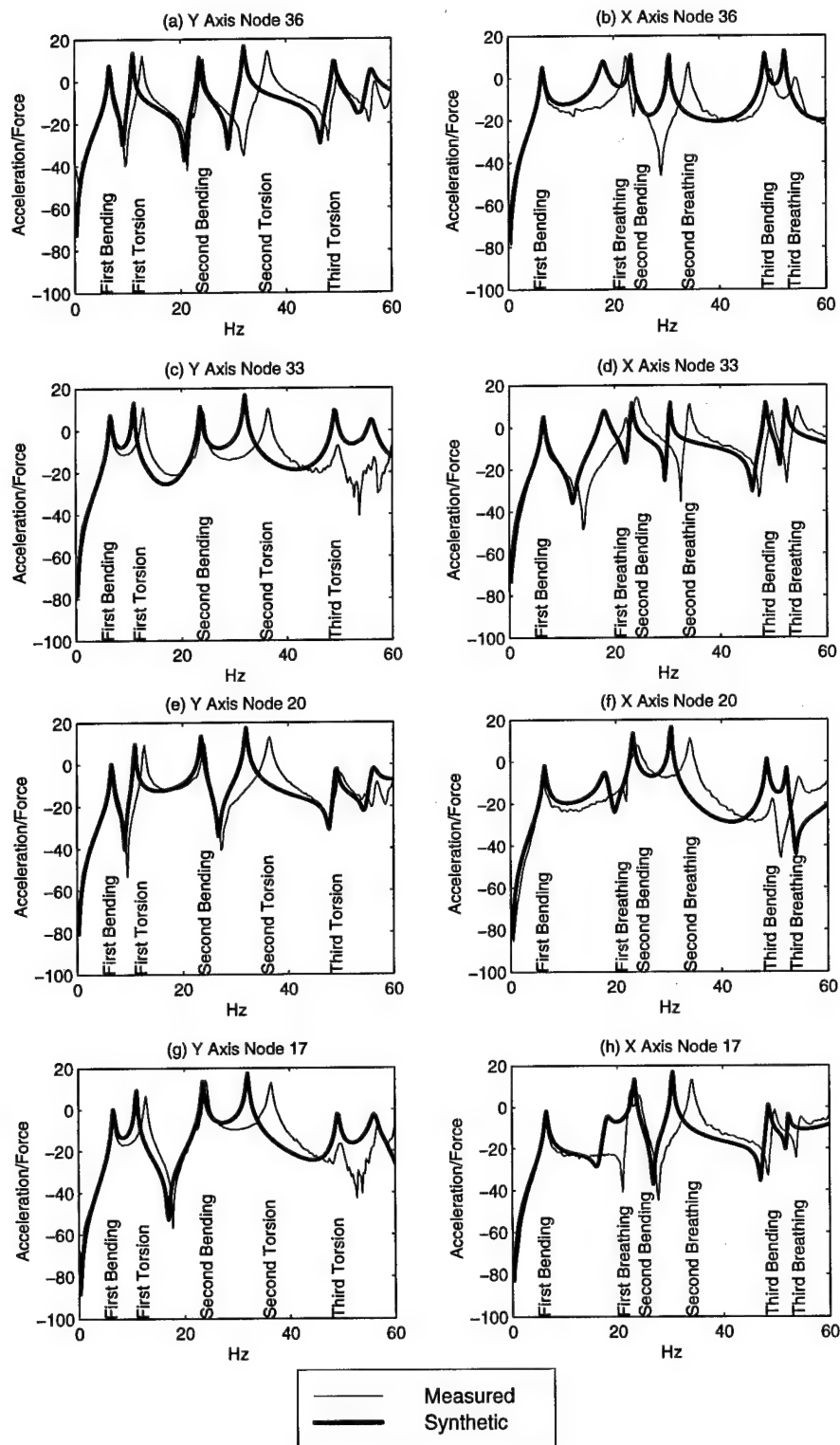


Figure 10. FTE baseline model results

3.4.2 Stiff Model

The first attempt to incorporate the additional stiffness of the vertical and horizontal plates was to add rigid nodes to the model (see Figure 11). A rigid node is a node that is displaced from a base node but has the same degrees of freedom. A transformation matrix was created and applied to the mass and stiffness matrices to transfer the degrees of freedom. The rigid-node model had 1056 degrees of freedom, but after transformation, the model had 216 degrees of freedom. The rigid node model gave better results than the untuned baseline, but it could not be tuned in the SDT because the transformation matrix required to transfer the degrees of freedom from the rigid nodes to the base nodes could not be easily incorporated into the tuning algorithm. So, a new high-order model, called the stiff model, was created that could incorporate the properties of the plates and be tuned in the SDT.

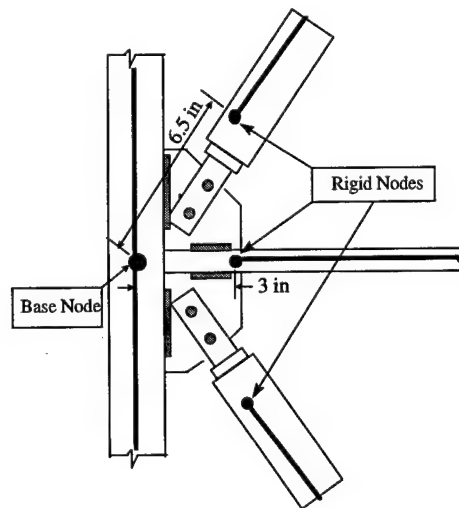


Figure 11. Diagram of rigid and base nodes

The purpose of the stiff model was to incorporate into the overall structure the stiffness provided by the vertical and horizontal plates (see Appendix A for properties) that connect the diagonals to the truss and to displace the lumped mass from the base node. The plates were modelled by adding in short stiff elements where the plates were. These new elements are referred to as “stiff” elements.

In Figure 12, one can see the different stiff elements added to the model. The total degrees of freedom increased to 996 for the stiff model. The increased tuning and processing time was traded for improved physical significance and correctness.

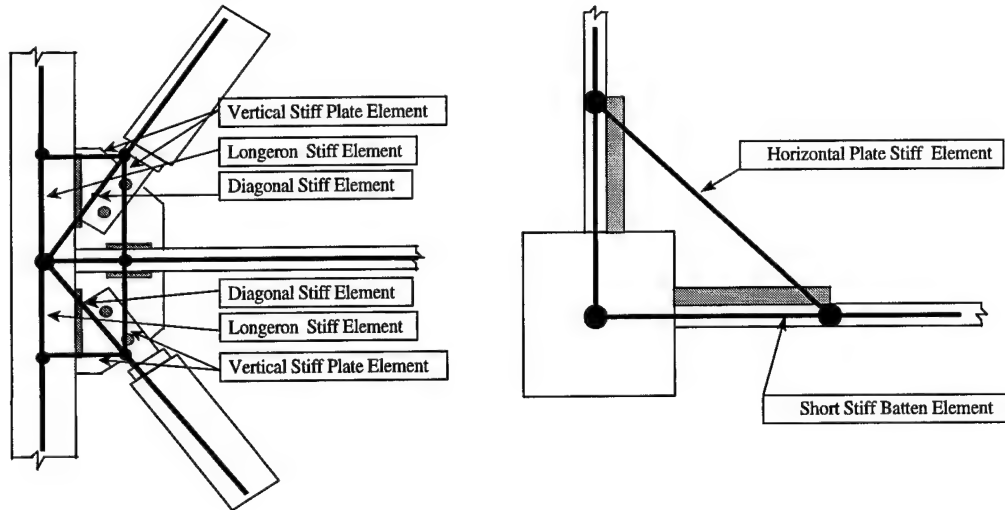


Figure 12. Diagram of nodes and stiff elements for vertical and horizontal plates

Since the stiff beam elements were approximations of the plate, the lengths were set to try and match the dimensions of the plate. The parameters (E, I_1, I_2, I_3, G, A , and ρ) of the stiff elements were estimated and adjusted to give a good initial match between the measured and predicted FRFs. Details such as the non-uniform cross section or two-bolt connection between the plate and short aluminum section made it very difficult to model correctly. The parameters for these short stiff elements were set as tuning parameters in the next section to try to improve physical significance and correctness of the model. The stiff model produced a better correlation between the model and the experimentally measured results as can be seen in Figure 14.

The finite element model was important to making the damage detection methodology work on the real structure. The key was having the model match the output in different damage states. Realistically, the model could not match the measured results exactly in all of the damage states due

to the many assumptions taken in the modelling process. The next chapter covers how the baseline and stiff models were tuned to make them more correct.

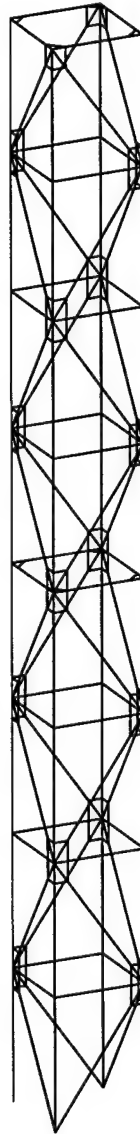


Figure 13. Diagram of the stiff model

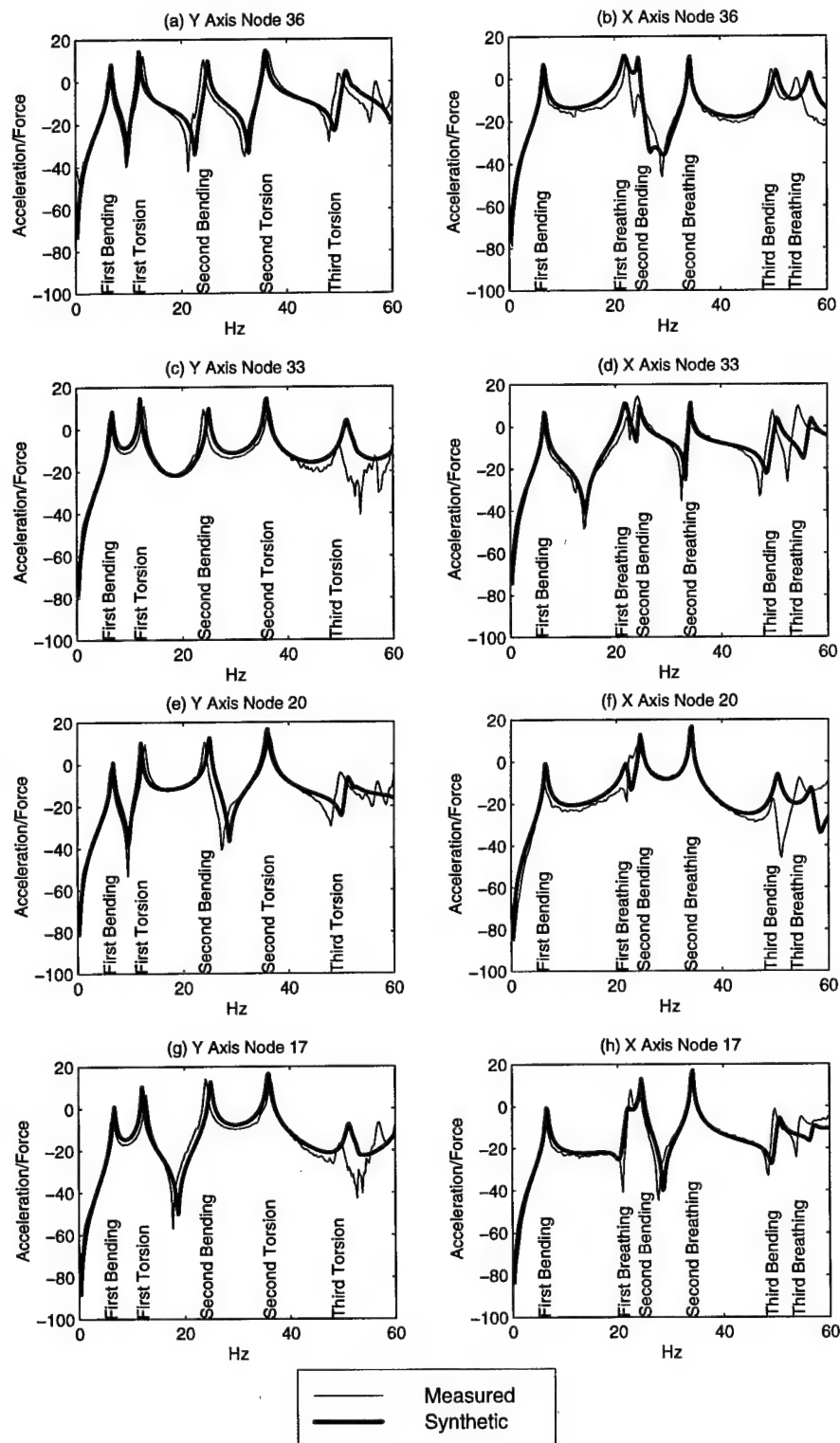


Figure 14. Stiff model results

Chapter 4 - Model Updating

4.1 Introduction

Updating of the finite element model was performed by selecting and adjusting parameters to increase the dynamic correlation between the response of the model and the measured data from the FTE. It was not an easy process, because many simplifying assumptions had to be made. The goal of the model updating process was to modify the baseline and stiff models to make it as correct as possible without losing physical significance. Model updating is also referred to as tuning in this thesis.

In 1993, Mottershead and Friswell [15] conducted a survey of the many the different types of methods of model updating. They broke the methods down into three categories: model identification, methods using modal data, and methods using FRF data. A method using FRF data directly was chosen for this research and compared to a model that was updated using modal data [4]. This chapter covers the selection of the method, frequency points, parameters, and penalty function. The chapter concludes with the tuning results from different models. The following section was developed from Mottershead and Friswell [8, 15].

4.2 Selection of a Method Using FRF Data

Of the three categories of model updating described by Mottershead and Friswell, only the methods using modal data or FRF data could be used. Model identification methods create correct models that are not physically significant, and the model had to have both of these qualities for the damage detection method used in this research. The model needed to be physically significant so that it could be modified by simulating broken members.

Of the remaining groups, there are advantages and disadvantages to each group of methods. The methods that use the modal data have the advantage of not requiring knowledge of the damping

ratios, whereas the FRF methods require the assumption of the values of the damping ratios and their incorporation into the model. The major disadvantage of using modal data directly was that the extra step of computing mode shapes was required which can add error to the model needlessly. Other advantages of FRF methods are that fewer sensors are required, computational time is shorter, models easily handle high modal densities, and more information is available.

In this research, the Structural Dynamics ToolboxTM (SDT) used the FRF data directly to perform model updating. In the following sections, the selection of frequency points and parameters are described in detail. In the last section, a full description of the penalty function is followed by the results of tuning the different finite element models.

4.3 Frequency Point Selection

In 1994, Lammens, Heylen, and Sas [12] researched the effects of frequency point selection and damping models. In their research they found the selection of frequencies and the way damping factors are taken into account were important factors in obtaining reliable updating results. They found the best set of frequency points in the range of interest should not be located at the resonance and anti-resonance points. They also found that the points of resonance and anti-resonance depended heavily on the damping information and believed these points were less reliable measurements. Lammens, Heylen, and Sas summarized their results by stating that frequency points away from the resonance peaks gave more stable updating parameters, the undamped FE model yielded a more stable updating process, and the proportional damping approach yielded acceptable models.

Modal damping was used on the baseline and stiff models. This damping model allowed the adjustment of individual peak heights. In the plots on pages 42 to 46, one can see that this model did not model the real damping behavior of the truss completely. For example, the height of a peak for a given modal frequency at one node may not match at the next node. This discrepancy suggests

that the real damping behavior of the structure is much more complicated than can be modelled by a simple proportional damping model. As shown in Figure 15, the tuning points were selected away from all resonance and anti-resonance points. There were 75 tuning points in the frequency range from 0 to 60 Hz. Since the total number of tuning points only had a minor impact on the time to tune the model, the number of points was somewhat arbitrary.

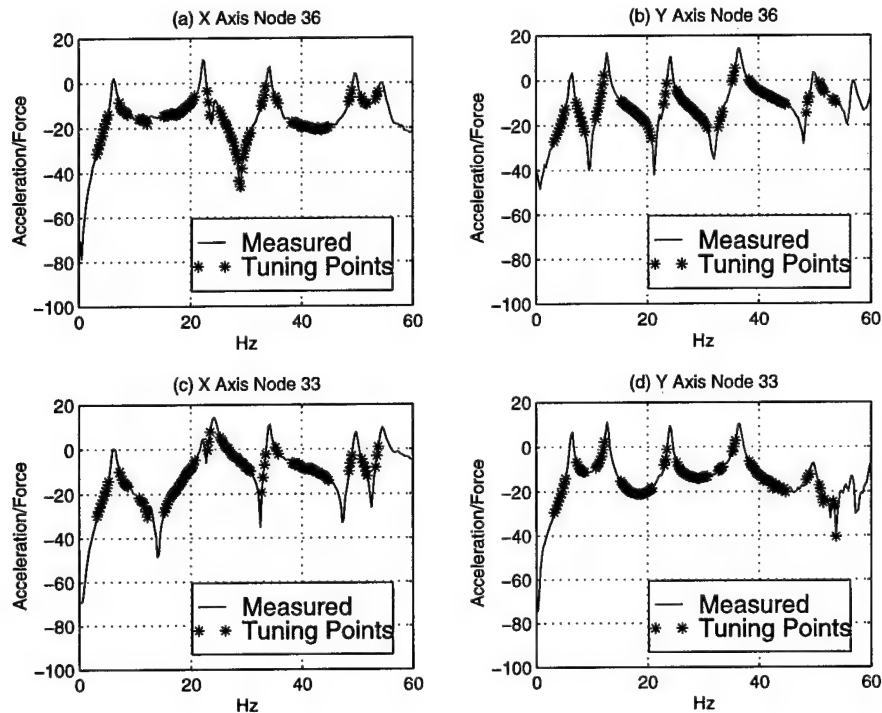


Figure 15. Tuning frequency points for nodes 33 and 36

4.4 Parameter Selection

Mottershead and Friswell [8] summarized that the number of parameters should be kept as small as possible. Also, such parameters should be chosen to correct areas of uncertainty in the model, and the FRFs should be sensitive to the parameters. The remainder of this section on parameter selection was developed from Balmes [1, 2].

After creating a physically significant model, parameters were selected for updating and adjusted so that the model had the correct response. In Appendix B, one can see that the element mass and stiffness matrices are dependent on E , G , ρ , A , I_1 , I_2 , J , and L . The matrices are linear in all the variables except for L . To keep the problem as linear as possible, linear tuning parameters β_j were created for substructures. The updated mass and stiffness matrices are defined by:

$$[M]_u = [M]_0 + \sum_{j=1}^k \beta_j [M]_j$$

$$[K]_u = [K]_0 + \sum_{j=1+k}^n \beta_j [K]_j$$

The $[M]_0$ and $[K]_0$ matrices are the mass and stiffness of the untuned models and $[M]_j$ and $[K]_j$ correspond to the substructure mass and stiffness matrices with β_j as the tuning parameter. The variable k is the number of tuning parameters for stiffness and the variable n is the total number of tuning parameters. The substructures were made from a collection of similar elements. For example, the β_1 parameter in the baseline model adjusted the longeron stiffness for all longeron members in the truss. The following tables list the substructures and tuning parameters used in the research for the baseline and stiff models.

Table 1. Substructures for tuning the baseline model

Parameter Number	Substructure
β_1	Battens Stiffness
β_2	Mid Battens Stiffness
β_3	Top Battens Stiffness
β_4	Longerons Stiffness
β_5	Diagonal Stiffness
β_6	Battens Mass
β_7	Mid Battens Mass
β_8	Top Battens Mass
β_9	Longerons Mass
β_{10}	Diagonal Mass
β_{11}	All Lumped mass from plates and connections

Table 2. Substructures for tuning the stiff model

Parameter Number	Substructure
β_1	Longerons Stiffness
β_2	Diagonal Stiffness
β_3	Diagonal Stiff Element Stiffness
β_4	Vertical Plate Stiffness
β_5	Top Batten Stiff Element Stiffness
β_6	Mid Batten Stiff Element Stiffness
β_7	Regular Batten Stiff Element Stiffness
β_8	Horizontal Plate Stiffness
β_9	Lumped Masses

The dynamic stiffness matrix can then be written as:

$$B(\{\beta\}) = [[M]_0 s^2 + [K]_0] + \sum_{j=1}^k \beta_j [K]_j + \sum_{j=1+k}^n \beta_j [M]_j s^2$$

Equation (1) can be rewritten as:

$$\left[[[M]_0 s^2 + [K]_0] + \sum_{j=1}^k \beta_j [K]_j + \sum_{j=1+k}^n \beta_j [M]_j s^2 \right] \{q\} = [b] \{u(s)\}$$

$$\{y(s)\} = [c] \{q\}$$

The predicted frequency response function $[H]_p(\omega)$ can then be calculated as:

$$[H]_p(\omega)_{NS \times 1} = \{c\} \left[[[M]_0 s^2 + [K]_0] + \sum_{j=1}^k \beta_j [K]_j + \sum_{j=1+k}^n \beta_j [M]_j s^2 \right]^{-1} [b] \{u(s)\}$$

The excitation force $[b] \{u(s)\}$ at $s = j\omega$ was white noise with a unity force but applied only to the degrees of freedom where the shakers were mounted. The resulting predicted FRF $[H]_p(\omega)$ matrix was 8x400. In the following section, this predicted FRF was compared to FRFs from the measured eight accelerometers and the 400 data points in the frequency spectrum from 0 to 100 Hz.

4.5 Penalty Function

A penalty function provides a measure of the degree to which the optimization of the model matches the measured data. Mottershead and Friswell [8, 15] described two different methods of error functions that the penalty functions are developed from: the equation error approach and the output error approach. The equation error approach was chosen because only a limited number of degrees of freedom could be measured. For more information about different approaches, see Mottershead and Friswell [8, 15]. The output error approach is a method to find the difference between the measured and predicted outputs. The penalty function was developed from the output error approach using a logarithmic scale.

The penalty function,

$$R = \sum \left| \log \left(\left[[H]_p(\omega) \right] \right) - \log \left(\left[[H]_m(\omega) \right] \right) \right|^2$$

is the square of the log magnitude difference between the measured $[H]_m(\omega)$ FRFs and the predicted $[H]_p(\omega)$ FRFs. The log magnitude was taken ($\log_{10}(\text{abs}())$) because the phase was found not to be useful in updating. Even though the mass and stiffness matrices were linear in the tuning parameters, this penalty function became a non-linear function of the tuning parameters β_j because the inverse of the dynamic stiffness matrix had to be taken. This non-linearity made the optimization problem much more difficult.

4.6 Optimization of the Penalty Function

After the parameters had been selected and optimization points had been chosen, an optimization algorithm had to be selected. The optimization algorithm determined the values of the parameters that minimized the penalty function within the parameters' allowable range. Through an iterative optimization routine, the tuning parameters in the penalty function were adjusted, and the error between the measured and synthetic FRFs was reduced. The disadvantage of this approach is

that it requires the minimization of a non-linear penalty function, which led to minor convergence problems and large computational time requirements. Additionally, the iterative scheme required the evaluation of the analytical model at every iteration. Balmes [1] found this function could have several local minimums resulting in non-unique solutions.

The SDT allowed 3 options for the optimization routine: simplex search routine algorithm of Nelder and Mead [17], a steepest descent method, and a quasi-Newton method using quadratic and cubic line search procedure. The three methods were evaluated, and the simplex search routine of Nelder and Mead gave the best results. Since the Lanczos algorithm was used for computing the FRFs at each step of the optimization process, only the modes between 0 and 60 Hz had to be computed. For example, only 20 of the possible 192 modes for the baseline or 20 of the possible 996 modes for the stiff model had to be extracted. The first 20 modes actually extended past 60 Hz, but this was done so the higher modes near 60 Hz would not be adversely affected. This reduced the time to compute one complete set of 8 FRFs from 8 to 2.5 seconds for the baseline model, and from 996 to 35 seconds for the stiff model which significantly reduced the optimization time.

Besides the location of tuning points, other factors such as damping, number of tuning points, step size, and starting point affected the tuning process. Several changes had to be made to the SDT tuning algorithm to make it work better. Modal damping was added and initial estimates of the damping factors had to be assumed. Additionally, the program had to be modified to compare acceleration data instead of displacements. This was done to stay consistent with the measured data.

As shown in Figure 15, there were a total of 75 tuning points for all 8 FRFs, and the points were all kept away from the resonance and anti-resonance peaks. The starting point for the optimization search had the greatest effect on the routine's ability to find a close solution. If the starting point was far away from the solution, the optimization routine took over 10 hours to complete. If the starting point was close, the routine only took 30 minutes or less. Initially, the starting points were

set as random variables between -0.5 and 0.5. Initially, 20 SPARC 20 computers ran the updating routine using random starting points and observed for convergence to similar final values. After a satisfactory methodology was developed, the optimization routines took about 200 minutes on the baseline model and about 900 minutes on the stiff model to converge to a solution. The following section contains the results of tuning (updating) the baseline and stiff models, and they are compared against the model tuning process using modal data.

4.7 Tuning Results

This section contains the tuning results from the baseline and stiff models. The results are compared to the modal data tuned model. The stiff model will be referred to as the stiff FRF-tuned model to indicate that it had been tuned using FRF data. The name for the baseline model after tuning using FRF data will be referred to as baseline FRF-tuned. The baseline modal data tuned model refers to the model developed by Cobb [4]. Since the plots of the results are each a page long, they were all placed at the end of this chapter. Table 3 contains the results of the penalty function measured at the tuning points. Since the modal data tuned model started from a similar baseline model, one can see the different tuning methods produced similar results. The modal tuned model was optimized to match measured modeshapes and modal frequencies. The rigid-node model was included to show that by simply adding in extra stiffness at the plate locations the model became more correct without tuning. The stiff-tuned model produced the best results for this penalty function.

Table 3. Penalty function values before and after tuning

Finite Element Model	Penalty Function Value
Baseline	110.3
Rigid-Node Untuned	39.9
Baseline Modal-Tuned	36.8
Baseline FRF-Tuned	26.6
Stiff	29.1
Stiff FRF-Tuned	18.4

4.7.1 Tuning Results from Baseline FRF-Tuned Model

To get the baseline model to match the measured data, the tuning process had to make large changes to the tuning parameters. These changes are indicated in Table 4. The FRFs of the baseline model before tuning are shown in Figure 16 on page 42, and the baseline FRF-tuned model FRFs are shown in Figure 17 on page 43. Before tuning, the first torsion, second torsion, first breathing, second breathing, and third breathing modes were below measured. The tuning process was able to make all of the peaks under 40 Hz match, but the third torsion and breathing shifted above the measured frequencies. These changes in modal frequencies are intuitively linked to the changes in the tuning parameters.

Table 4. Tuning parameter results from FRF tuning of the baseline model

Parameter Number	Substructure	Value
β_1	Battens Stiffness	0.776
β_2	Mid Battens Stiffness	1.107
β_3	Top Battens Stiffness	1.081
β_4	Longerons Stiffness	-0.219
β_5	Diagonal Stiffness	0.578
β_6	Battens Mass	1.732
β_7	Mid Battens Mass	0.694
β_8	Top Battens Mass	0.069
β_9	Longerons Mass	0.089
β_{10}	Diagonal Mass	1.039
β_{11}	All Lumped mass from plates and connections	0.001

The tuning algorithm increased the batten stiffness parameters β_1 , β_2 , and β_3 to increase the breathing mode frequencies. These increases were significant because they were trying to incorporate the missing plate stiffness. Unfortunately, increasing the stiffness of the entire batten makes the model less physically significant even though it makes it more correct. The increase in diagonal stiffness β_5 directly increased the frequencies of all of the bending and torsion modes. The bending and torsion modes were decreased by the decrease in longeron stiffness β_4 . Finally, the mass mo-

ment of inertia of the structure was significantly increased by increasing all of the mass parameters. This increase in mass decreased the frequencies of the bending and torsion modes.

Overall, many adjustments were made and the net effect was to get the frequencies to match as closely as possible on the available time. For this baseline model case the optimization routine did its best with the variables it had. A closer match might have been realized if the number of parameters had been increased.

4.7.2 Tuning Results from Stiff FRF-Tuned Model

Since the stiff model started out with a lower penalty function value than the baseline model, the tuning parameters did not have to be adjusted much. Their final values are shown in Table 5. The FRFs of the stiff model before tuning are shown in Figure 19 on page 45, and the stiff FRF-tuned model FRFs is in Figure 20 on page 46. Before tuning the stiff model the penalty function was 29.1, and after tuning the penalty function was 18.4. The reason it was so small initially was the initial values of the stiff elements added to the model were set to make the initial model have a close match and let the tuning process find the best value.

Table 5. Results from FRF tuning of the stiff model

Parameter Number	Substructure	Value
β_1	Longerons Stiffness	-0.21128
β_2	Diagonal Stiffness	-0.075131
β_3	Diagonal Stiff Element Stiffness	0.17976
β_4	Vertical Plate Stiffness	-0.30580
β_5	Top Batten Stiff Element Stiffness	2.5137
β_6	Mid Batten Stiff Element Stiffness	-0.79356
β_7	Regular Batten Stiff Element Stiffness	0.23113
β_8	Horizontal Plate Stiffness	-0.096127
β_9	Lumped Masses	-0.18495

The physical interpretation of these values is less intuitive because stiff beam elements were used to model the plate stiffness. The tuning process made adjustments to these initial values to get

the fit. The key to tuning the stiff model was making adjustments to the initially estimated “stiff” beam elements because the plate cannot really be modelled accurately by a single beam because it has a varying cross section and with a bolted connection. In the next chapter, the baseline FRF-tuned, baseline modal-tuned, and stiff FRF-tuned models are used to train various pattern classifiers to detect damage in the FTE.

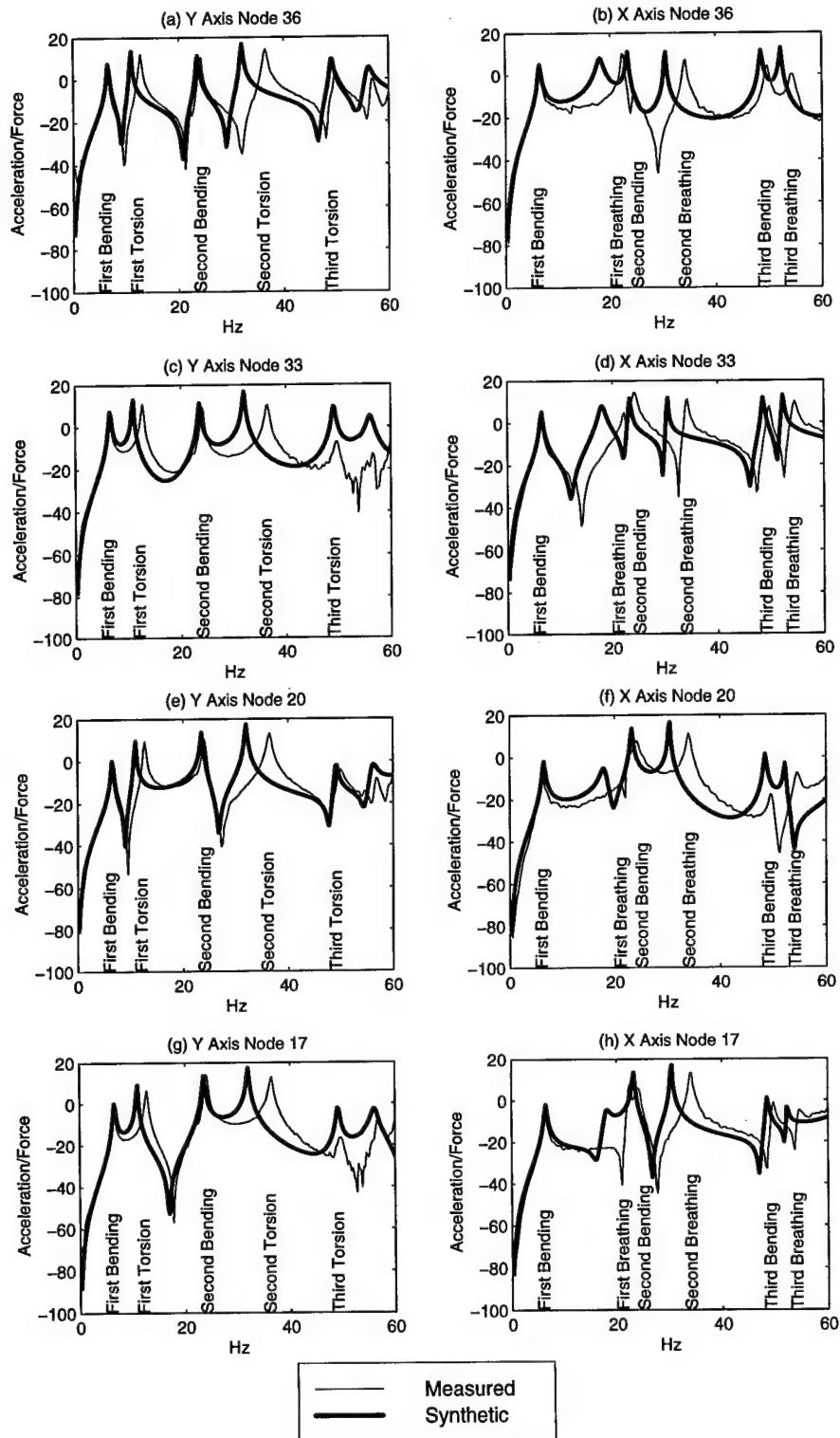


Figure 16. FRFs from baseline model before tuning

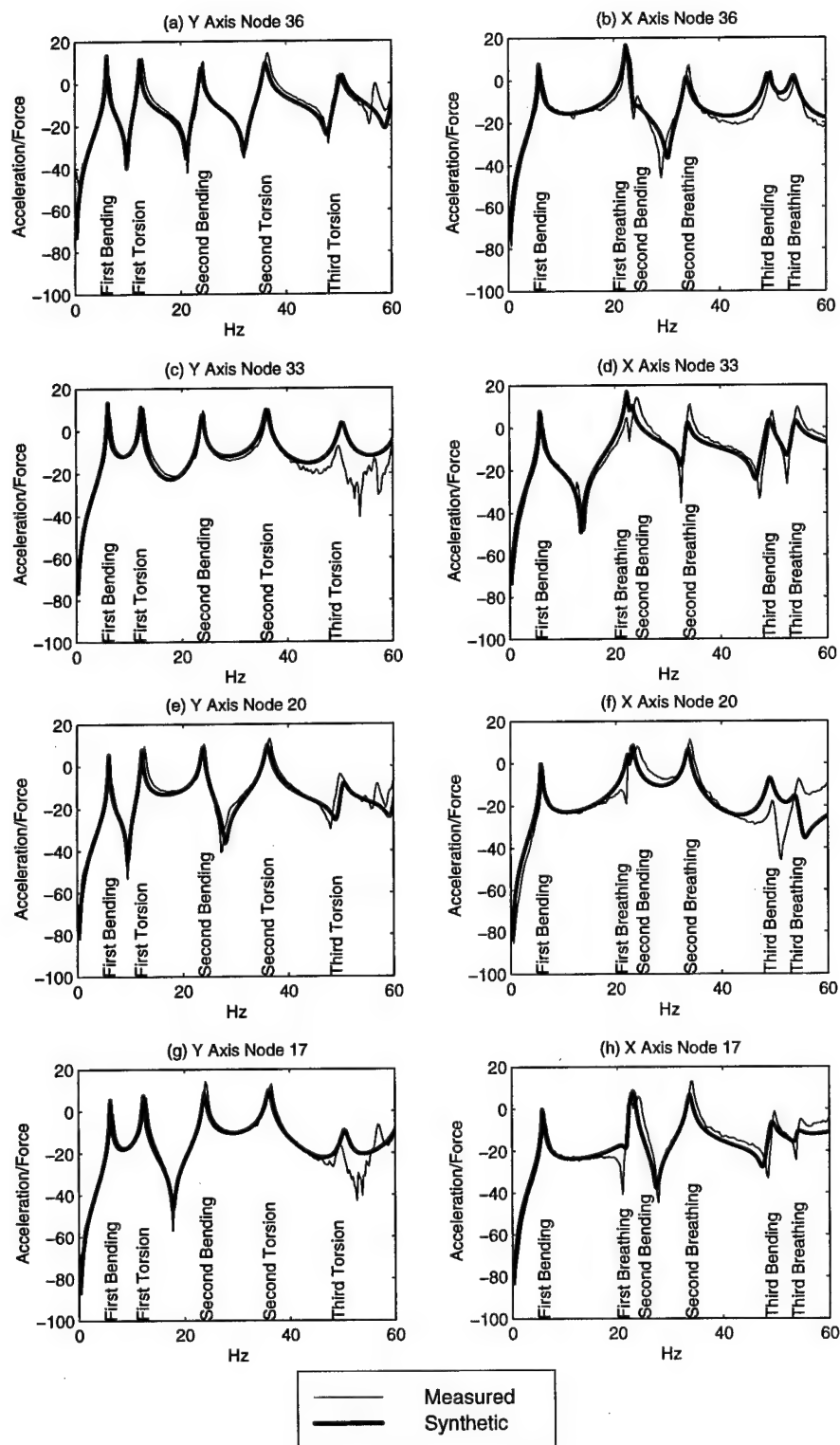


Figure 17. FRFs from baseline after tuning using FRFs

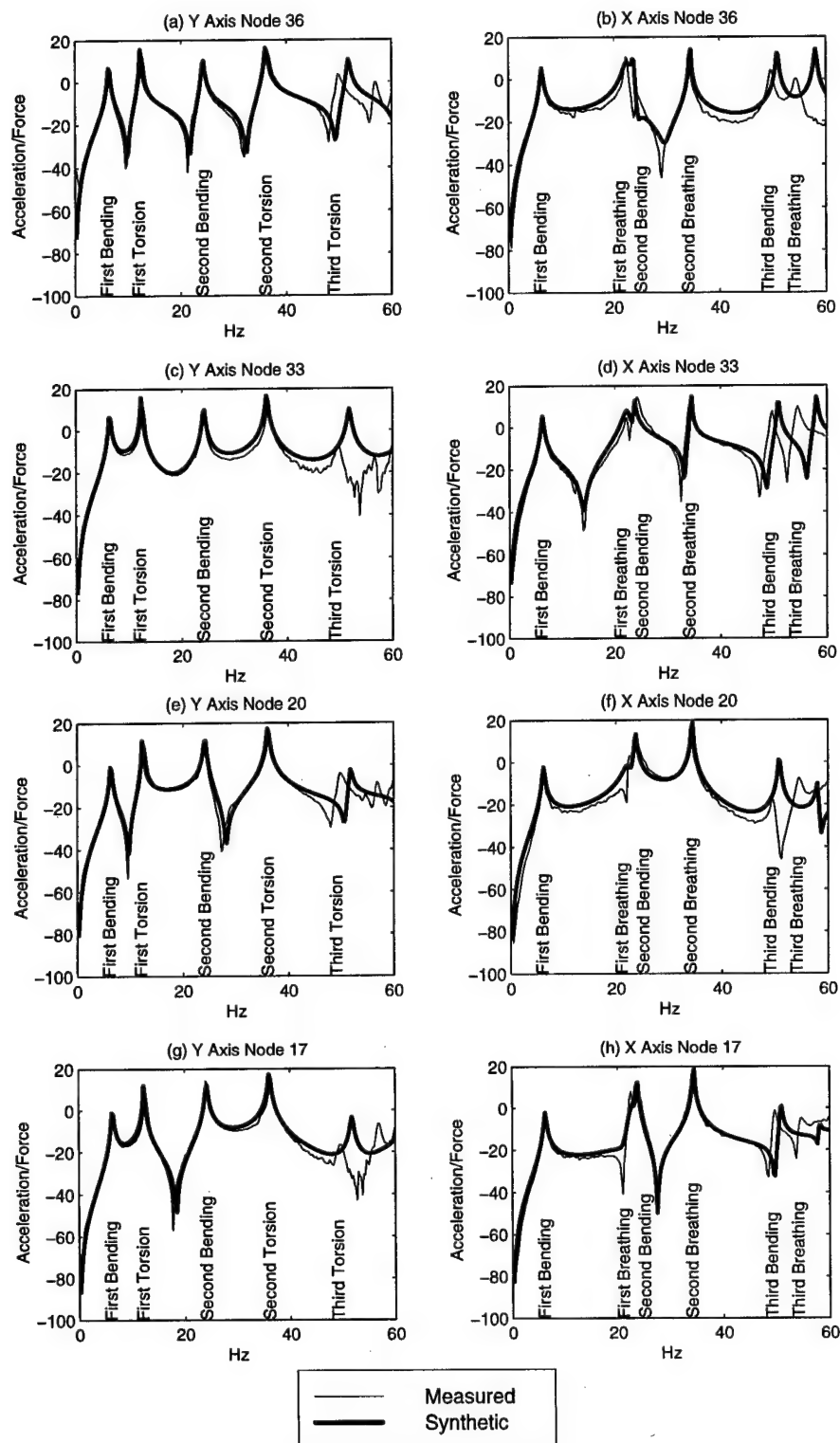


Figure 18. FRFs from baseline after tuning using modal data

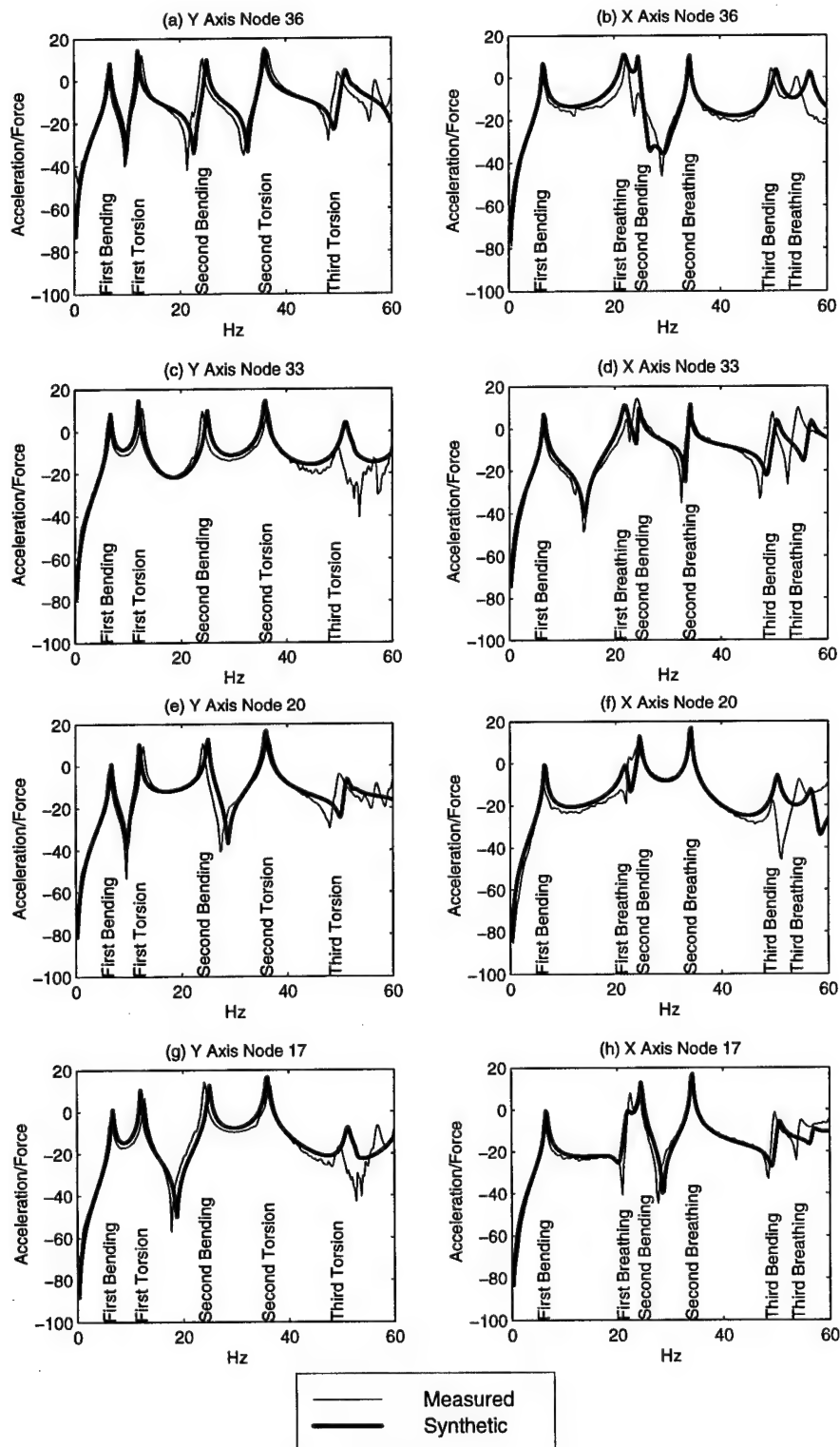


Figure 19. FRFs from stiff model before tuning

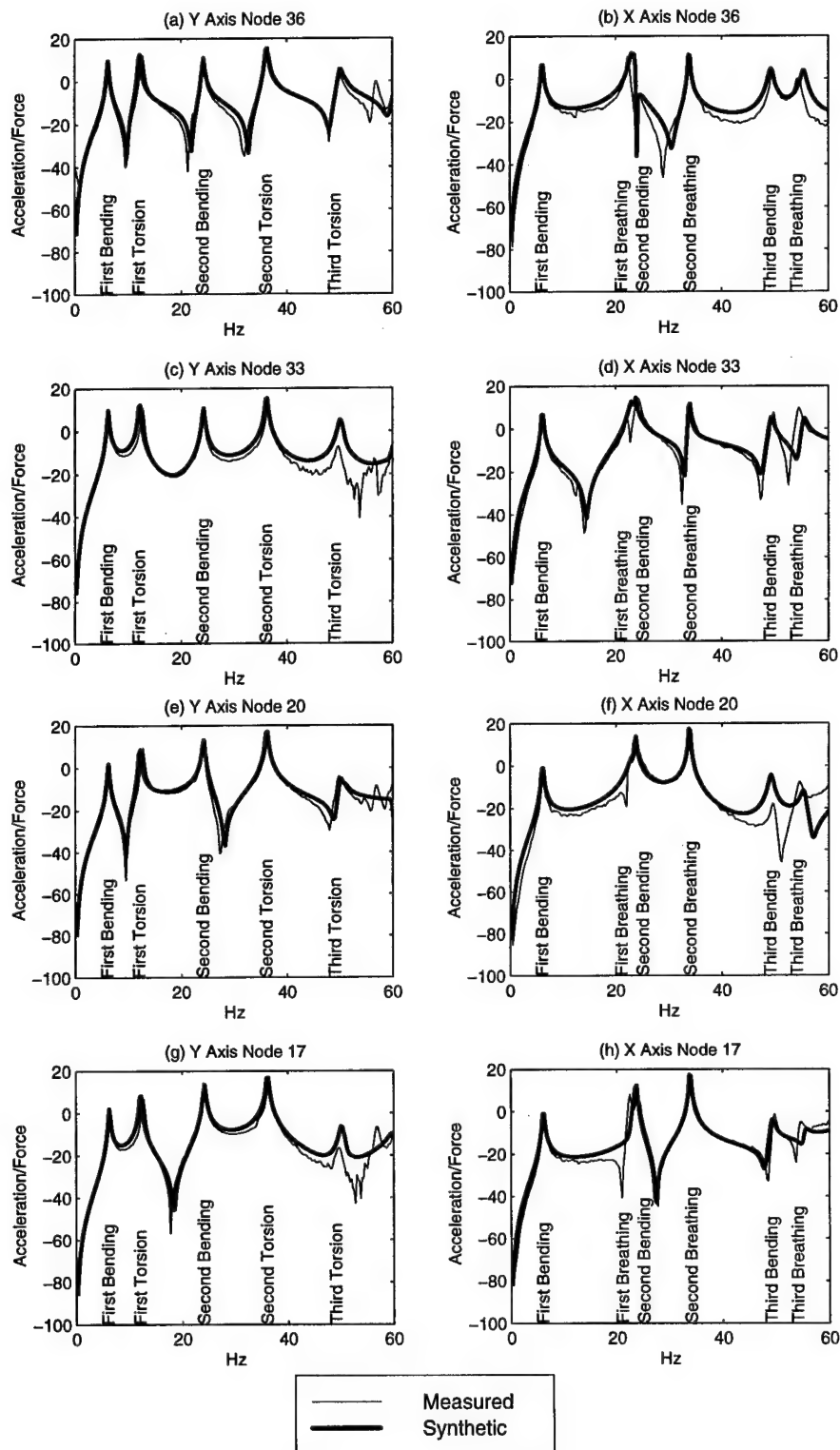


Figure 20. FRFs from stiff model after tuning

Chapter 5 - Pattern Classifiers

5.1 Introduction

The differences in vibration characteristics of the various damage states were extracted from the real time accelerometer output through a process called feature extraction. The feature extraction process was used to reduce the data by measuring certain “features” or “properties” that distinguished the different damage states. The feature extractor in this experiment converted the real-time accelerometer data from two accelerometers into a 20 element feature vector. The feature extractor can be thought of as a property filter or attribute detector.

These feature vectors were then passed to a classifier and either used for training or classifying. Classifier training is the process where by many patterns are used to define the features of each damage class [19, 51]. The damage classes are the different types of damage. Pattern classification is the process of figuring out which class an unknown pattern belongs to. This chapter first covers the variation in the measured FRFs and how the feature vectors were extracted from these FRFs. Next, the pattern classifiers are described followed by their results and an explanation of results.

5.2 Data Variation

The measured FRF data was collected with only 100 averages using maximum overlap. This resulted in the measured data looking quite variable as seen in Figure 21(b), where all of the 100 averaged measurements are shown. The number of averages was kept low so the measured data would have some variability in it to provide to the classifier a region of acceptability for each class.

A histogram, shown in Figure 21(d), was generated to show the variation in the measured data. One can see the data looks approximately normally distributed and was assumed to be so. Additionally, the variation in the data was tested for correlation and was assumed to be uncorrelated. The variation in the data was measured for each axis and used with a normally distributed number

generator to generate FRFs with similar distribution. In Figure 21(a), one can see 100 generated FRFs with variability added using the normally distributed random number generator. Figure 21(c) shows the histogram of the distribution from the normally distributed random number generator.

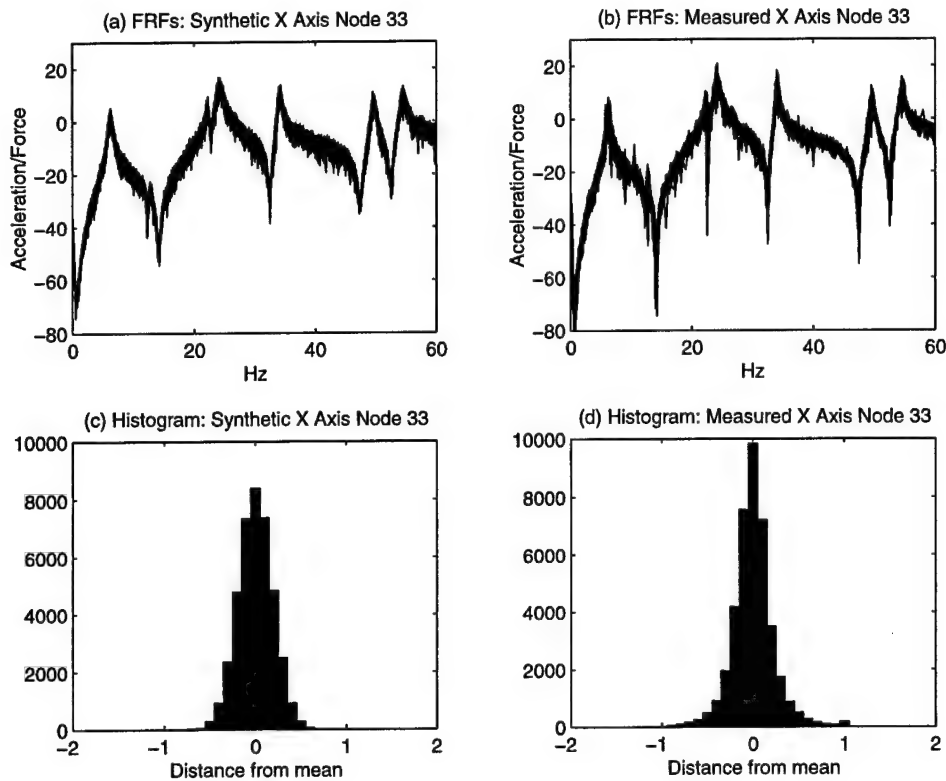


Figure 21. 100 FRF measurements overlayed with respective histograms of the variation in data

By recreating similar variance in the data using a normally distributed random number generator, the time it took to create a set of well distributed random set of training data from one synthetic FRF was about 3 seconds on a 133MHz Pentium processor. Even though this process was not the best solution in recreating the data similar to the measured sets, it was very quick and gave good results.

5.3 Feature Extraction

The FRFs looked very similar to the spectra of human speech. This observation led to adapting a feature extraction process commonly used in speech analysis. The feature extraction process started with converting the time domain data into the frequency domain using FFTs, as seen in Figure 22. Then, FRFs were computed (see Chapter 3 for details). The feature extraction process continued with passing the spectral data through a triangular filter bank.

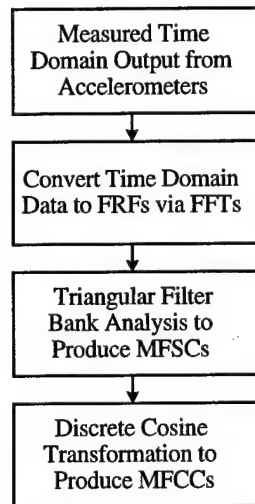


Figure 22. Block diagram of feature extraction process

The triangular filters were spaced linearly from 0 to 50 Hz and logarithmically from 50 to 100 Hz. The area under each triangular filter was 1 and used a weighting scale in a discrete integration process. A point by point multiplication and summation was performed for each triangle and FRF. Since there were 10 triangles per FRF, this reduced two 400 data point FRFs to a 1-by-20 vector called the Mel-Frequency Spectral Coefficients (MFSCs). Two accelerometers were used to increase the chances that the feature vectors were unique for each damage state. This vector was then transformed onto a space defined by a basis of orthogonal cosines to produce a set of coefficients called Mel-Frequency Cepstral Coefficients (MFCCs). This projection process both decorrelates and or-

thogonalizes the resulting MFCCs. This process was adapted from the Mel-Frequency analysis used in speech processing. Since the method used in this research was an adaptation, the names MFSC and MFCC were not precisely used (because they are usually defined on the frequency ranges found in speech) but the names were kept for clarity.

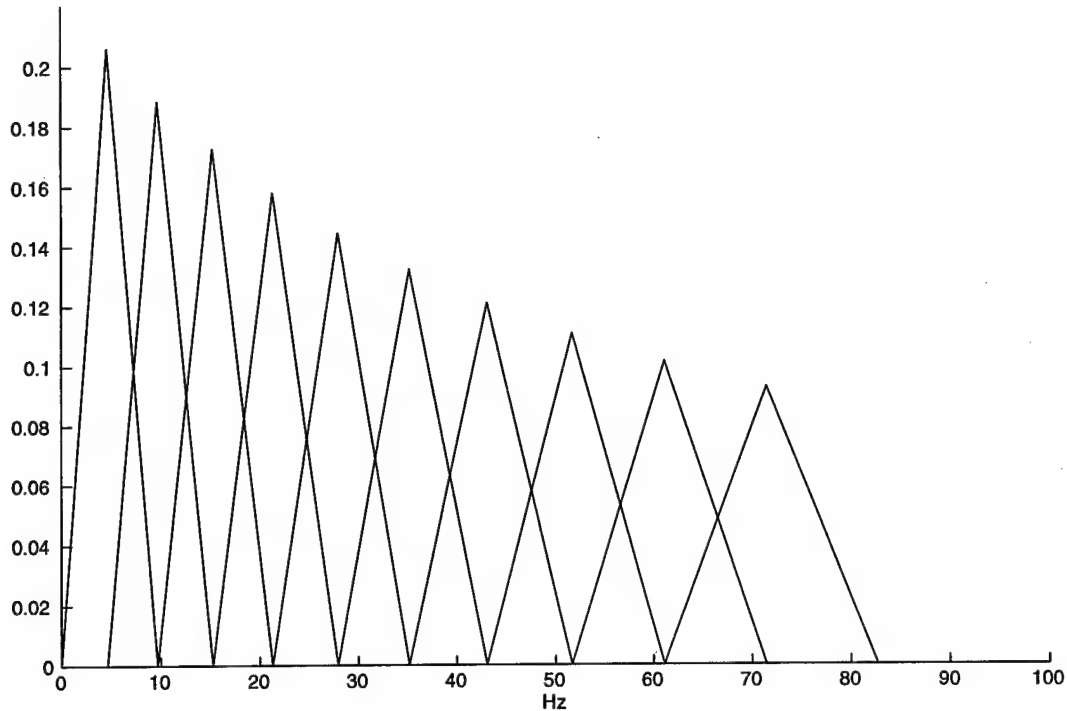


Figure 23. Diagram of triangular filter bank

In section 5.2, the variation in the FRFs was shown to be quite variable. After the FRFs were converted to feature vectors, they were used to either test or train the pattern classifiers. The feature vectors created a 20-dimensional space where each class tended to cluster in regions. Since it is impossible to draw a 20-dimensional space, a 3-dimensional example is provided from the measured test vectors. Figure 24 shows a 3-dimensional depiction of how the data is clustered. The three axes are the first three elements of the measured test vector. The pattern classifiers define these regions for each class and then test unknown vectors to determine which class the vector belongs to.

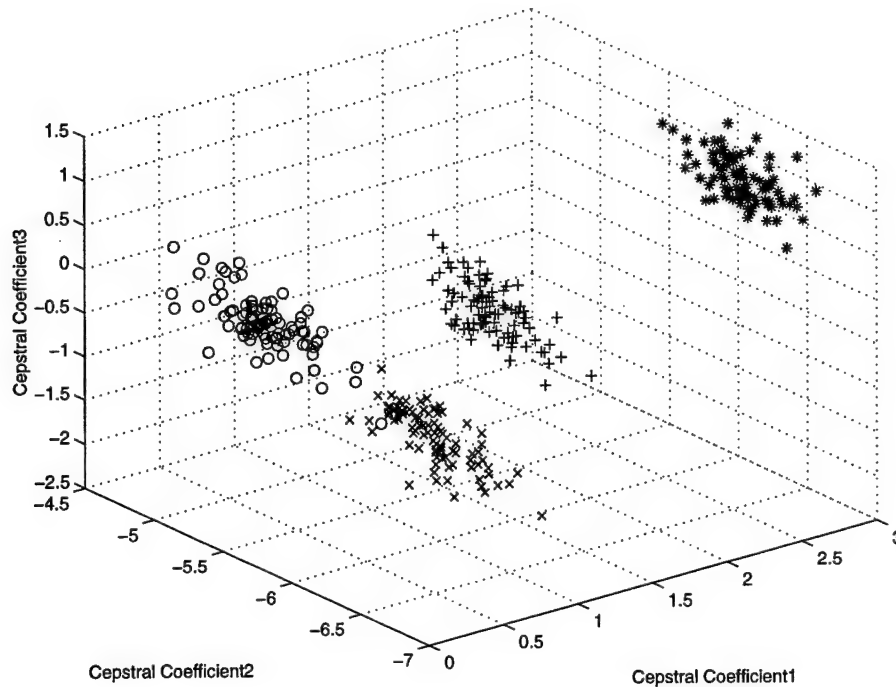


Figure 24. Example of measured test vector in 3-dimensional space

5.4 Pattern Classifiers

The two types of pattern classifiers used in this research were parametric and non-parametric. The Gaussian classifiers used are parametric because the mean and covariance of the training vectors were calculated and stored. The ANN is a non-parametric classifier because the features of the training data define the regions of each class, and the regions are used for classification.

5.4.1 Gaussian Classifiers

This section provides a quick overview of the Gaussian classifier and was adapted from Duda and Hart [6, 1-32]. In Duda and Hart's book, the Gaussian classifier is referred to as a particular type of Bayes classifier. This classifier is based around a decision theory where the decision of which class a particular feature vector belongs to is formed in probabilistic terms, and that all relevant

probability values are known. These relevant probability values are developed from the training data. The multivariate normal density is written as:

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp \left[-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^t \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right]$$

where \mathbf{x} is a d -component test vector, $\boldsymbol{\mu}$ is the d -component mean vector, Σ is the d -by- d covariance matrix, $(\mathbf{x} - \boldsymbol{\mu})^t$ is the transpose of $(\mathbf{x} - \boldsymbol{\mu})$, Σ^{-1} is the inverse of Σ , and $|\Sigma|$ is the determinant of Σ . The covariance matrix is computed as:

$$\Sigma = E [(\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})^t]$$

where $E[\]$ is the expected value operator and the mean is computed as:

$$\boldsymbol{\mu} = E[\mathbf{x}]$$

The ij th components of the covariance matrix can be found from:

$$\sigma_{ij} = E[(x_i - \mu_i)(x_j - \mu_j)]$$

and the i th component of the mean can be found from:

$$\mu_i = E[x_i]$$

The diagonal element of the covariance matrix σ_{ii} is the variance of x_i , and the off-diagonal element σ_{ij} is the covariance of x_i and x_j .

The Gaussian classifier was the pattern classifier used, because it produces the simplest separating hypersurfaces for classification and is straightforward to train and test. There were four variations of the Gaussian classifier tested. Table 6 describes these variations.

Table 6. Types of Gaussian classifiers

Pattern Classifier Name	Description, Discriminant Function $g(\mathbf{x})$, and Geometry
Grand Diagonal	$\sum_i = \sigma^2 I$, One diagonal covariance matrix for all classes $g_i(\mathbf{x}) = -\frac{1}{2\sigma^2} \ \mathbf{x} - \boldsymbol{\mu}_i\ ^2$ Geometric Interpretation: hyperspherical clusters
Grand Full	$\sum_i = \sum$, One full covariance matrix for all classes $g_i(\mathbf{x}) = -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_i)^t \sum^{-1} (\mathbf{x} - \boldsymbol{\mu}_i)$ Geometric Interpretation: hyperellipsoidal clusters
By Class Diagonal	$\sum_i = \sigma_i^2 I$, Diagonal covariance matrix for each class $g_i(\mathbf{x}) = -\frac{1}{2\sigma_i^2} \ \mathbf{x} - \boldsymbol{\mu}_i\ ^2$ Geometric Interpretation: hyperquadratics
By Class Full	$\sum_i = \sum_i$, Full covariance matrix for each class $g_i(\mathbf{x}) = -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_i)^t \sum_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i)$ Geometric Interpretation: hyperquadratics

The discriminant function $g_i(\mathbf{x})$ corresponds to a distance measure between the test vector \mathbf{x} and the mean $\boldsymbol{\mu}_i$ of each class. The class with the smallest discriminant function is selected. A simple way of looking at the problem is each class has a mean vector that is the ideal template for each class. The distance between the test vector and the mean vector is measured using either the Euclidean norm $\|\mathbf{x} - \boldsymbol{\mu}_i\|^2$ or squared Mahalanobis distance $(\mathbf{x} - \boldsymbol{\mu}_i)^t \sum_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i)$. In the following section, the results from each Gaussian classifier are presented.

5.4.2 Artificial Neural Networks (ANNs)

There are numerous references on ANNs, however this quick overview on ANNs was adapted from Rabiner and Juang [19]. ANNs are computational models inspired by neuron architecture of the human brain. Using ANNs, one can create a non-linear transformation between arbitrary inputs and outputs. In this research, a multilayer layer perceptron (MLP), which is a type of ANN, was used as a transformation between feature vectors and the damages states.

The MLP topology is complex and is defined in layers of neurons defined as hidden layers (see Figure 25). The input and output layers feed the information through from left to right.

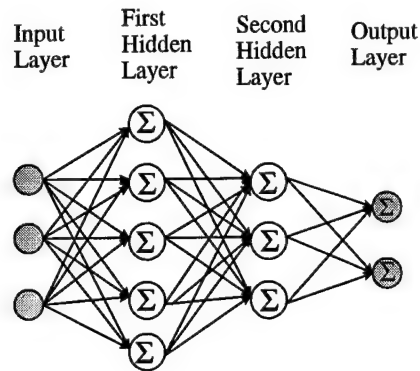


Figure 25. Diagram of a typical MLP

In the 20-dimensional feature space, the classes are divided by hypersurfaces. The ANN will divide the 20-dimensional space into regions by these hypersurfaces, and each class is contained in one or more of these regions. These regions are defined during the training process. During the classification phase, the feature vectors were tested to see which region they fall into. One can visualize from Figure 24 the hypersurfaces containing each cluster of data.

The MLP used in this thesis was developed and tested in a program called LNKnet. Several variations were tested, and the MLP configuration that gave the best results was used. This MLP had 20 input nodes, 25 hidden nodes, and 33 output nodes. It had only one hidden layer and the activation function was a standard sigmoid in the hidden and output layers. The MLP had to go through about 30 epochs in training before the error rate was less than 0.01.

Due to widespread testing using MLPs, it was tested and compared to the Gaussian classifiers. The best MLP was tested, and the Modal Data Tuned Model had 46% error rate on 3,300 measured test vectors, whereas the baseline FRF-tuned model had a 61% error rate on 3,300 measured test vectors. Since these results were not better than the Gaussian Classifiers, the MLPs were not used

any further in the research. The remainder of the research focused on variations of the Gaussian classifier.

5.5 Damage Detection Results

Initially, the pattern classifiers were tested using measured training data and measured test data. All five of the classifiers used in this thesis had no errors when trained with 1650 measured training vectors and 1650 different measured test vectors from a total of 33 classes. From this test, the correct selection of feature extraction method and pattern classifier was validated. If all of the pattern classifiers could distinguish every test vector without error that meant the features for each damage state were unique and identifiable. This also confirmed that variables such as number of triangles in the triangular filter bank and number of accelerometers did not have to be questioned. What had to be questioned was whether or not the same results be realized if the classifiers were trained using data generated by the FE models (called synthetic data).

As stated before, the entire purpose of generating a good FE model of the FTE was to have the models produce FRFs from different damage states. The damage was modelled in the structure by a reduction in stiffness in each of the diagonal members. With 32 diagonal members and a healthy state, there were 33 damage states or classes to be tested. The tests were initially done with 100 training vectors per class, but the results varied. Finally, the training was performed with 300 training vectors from each class, and the results did not vary. The accuracy rates are tabulated in Table 7

Table 7. Accuracy rate calculated from 300 synthetic training vectors from each class and 100 measured test vectors from each class results

FE Model vs Classifier	Baseline Modal-Tuned	Baseline FRF-Tuned	Stiff FRF-Tuned
Grand Diagonal (G-D)	80%	62%	66%
Grand Full (G-F)	78%	66%	79%
By Class Diagonal (BC-D)	82%	64%	77%
By Class Full (BC-F)	73%	66%	74%
Average	78.25%	64.5%	74%

5.6 Accuracy Rate Analysis

Since it was shown that the classifiers had sufficient features to distinguish each class, the source for the low accuracy rate came from the FE model's inability to model the different damage states correctly. Mottershead and Friswell [15] summarized the causes of the inaccuracies or differences between the measured system and model prediction as being caused from: model structure, parameters, model order, testing, noise, or measurement. This section considers the sources of error in detail and points out areas for improvement.

Model structure errors can be caused by uncertainty or misapplication of the governing physical equations of the finite element method. For example, beam elements were used to model sections of the vertical and horizontal plates as an approximation to keep the total number of degrees of freedom as small as possible. Model parameter errors including incorrect parameter measurements, boundary conditions, and joint conditions probably caused most of the model inaccuracies. The joints with the plates caused most of the modelling difficulties. The tuning process made adjustments to these properties, but there is always room for error. Model order errors were caused by the discretization of the structure being too small. The model order errors was addressed in the stiff model. Even though the stiff model had a large increase in DOFs from 216 to 996 the model could still be discretized much further.

There were many sources of error in experimental testing. Changes in the dynamics of the structure caused by the dynamics of the equipment used to excite or measure the structure might not have been accounted for correctly. Noise in the measuring and excitation equipment was an undetermined source of error. Errors from processing the vibration measurements could have arisen from aliasing or spectral leakage.

One of the unexpected sources of error came from local modes created by broken members. Unexpected peaks showed up in the damaged structure's FRFs that could not initially be accounted for. After careful analysis, it was found that the excited disconnected members on the FTE showed up in the FRFs as a local modes. The test data for the damage states was generated by disconnecting one end of a diagonal, whereas the synthetic data was generated by zeroing out the diagonal member's stiffness. This was not an exact model of the damage. In Figure 26 on page 59, one can see the local modes in the Y axis at 12.25 Hz and the X axis at 21 Hz. Since these local modes were not modelled into the damage simulation in the models, this became a significant source of error. One can see from the plots from each of the models that these local modes did not show up in the models. More detailed analysis would be required to incorporate these modes into the models. Additionally, the damage of any member could happen in various ways. For example, if the member broke completely or partially at any point in the member the local modes would be distinctly different. To model any type of break measurements, each type of break would have to be measured and then these breaks could be lumped together as one damage class.

Initially, this damage detection process used two accelerometers at node 36. One accelerometer was parallel to the shaker and the other orthogonal. The orthogonal accelerometer did not match up well because of the complex damping characteristics of the structure. It was discovered that using the accelerometers directly under the shakers and in the same orientation gave superior results. There was about a 10% decrease in the error rate for all pattern classifiers for all 33 damage states.

Finally, the misclassification errors were carefully examined and broken into three groups. The misclassification errors tended to happen in bays or the member directly above or below the broken member. This group was called the related member error group. The other types of misclassification resulted from complete misclassification, which was rare, and classification as the unbroken state. The unbroken state misclassifications were only seen at the top of the truss and were probably a result of the this type of damage not causing a very large change. Table 8 shows how the error rate broke down for the by class diagonal Gaussian classifier for each of the models. Clearly the baseline modal-data tuned model gave the best results because it had the majority of its error rate in the related members category. If the goal was to identify the member, or the related member the success rate would have been 97% for the baseline modal-tuned model, 88% for the baseline FRF-tuned model, and 86% for the stiff FRF-tuned model. The related member error group was very similar to Cobb's small subsection of the truss. He was able associate different damage states before damage detection and his methodology was able to classify the damage two the correct subsection in two tests.

Table 8. Error rate from By Class Diagonal

By Class Diagonal Classifier	Baseline Modal-Tuned	Baseline FRF-Tuned	Stiff FRF-Tuned
Initial error rate	18%	36%	23%
Related member errors	15%	24%	9%
Unbroken errors	2%	11%	10%
Unrelated errors	1%	1%	4%

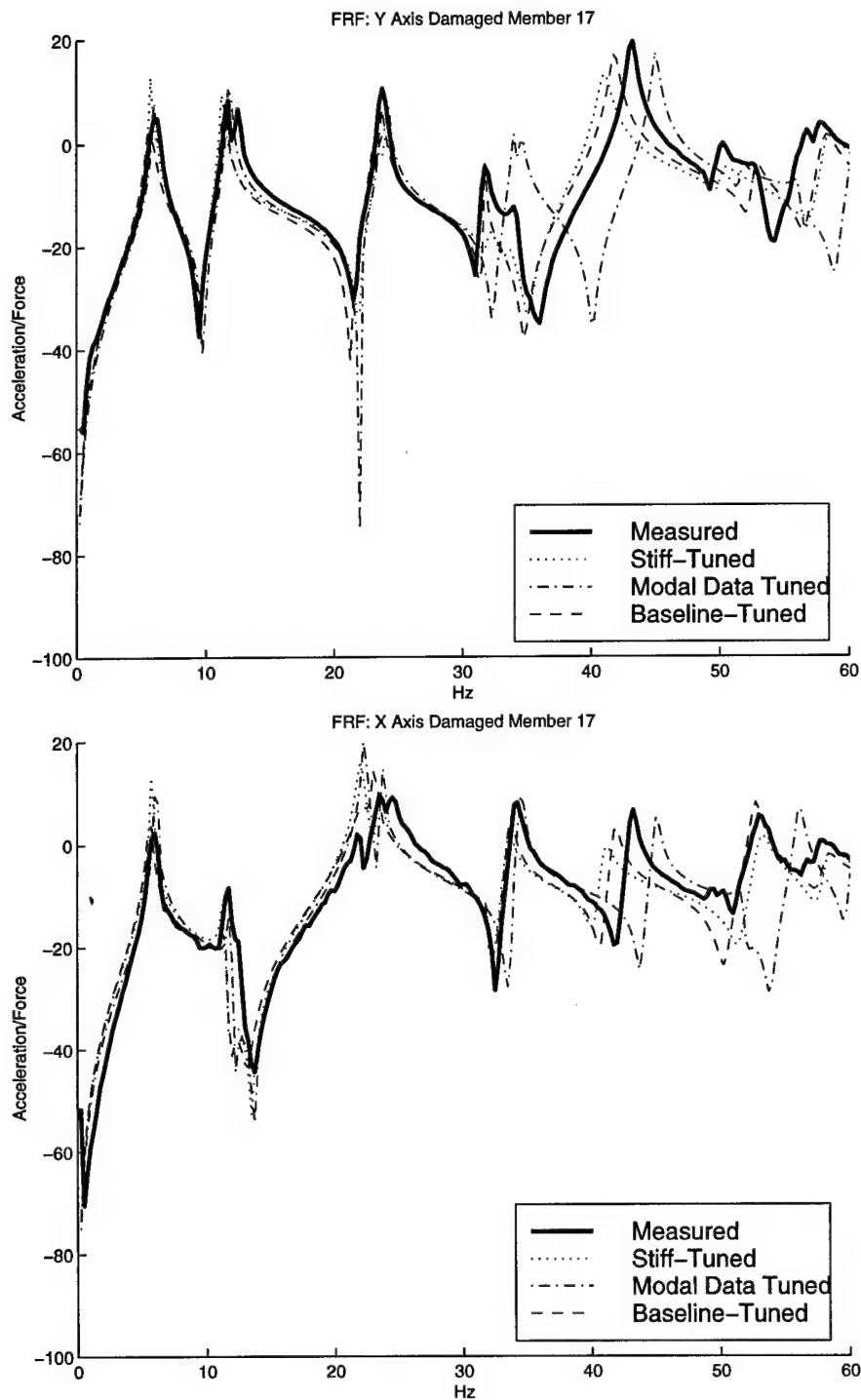


Figure 26. Damaged state showing measured and synthetic FRFs

Chapter 6 - Results and Conclusions

6.1 Compilation of Results

Since the research covered several different areas, several different contributions were made. The research first contributed to the on-going characterization of the FTE. Several sources of errors were found in previous finite element modelling attempts, and the solution provided made improvements. Direct use of FRFs for model updating was tested and found to give similar results as a method that used modal data. The research showed that unique feature vectors can be extracted from FRFs that allow a pattern classifier to uniquely identify the damage states. Attempts were made to get the FE models to match the measured data results. The following is an overview of the results found in each chapter with additional comments. For more details, refer to the appropriate chapter.

6.1.1 Finite Element Model Development Results

Previous researchers who developed finite element models of the FTE had problems getting the models to accurately predict the measured response of the FTE. Their tuned models typically had trouble with the torsion and breathing modes having too low of a modal frequency. In this thesis, a baseline model was created using the results from a previous model using the Structural Dynamics ToolboxTM for MATLABTM. This baseline model had the same problems as the others, and the source of the problem was found to be the lack of correctly modelling the welded plates. A high order model called the stiff model was created to include the stiffness and mass displacement added by the horizontal and vertical plates. This greatly improved the prediction of virtually all of the modes in the 0-50 Hz range.

6.1.2 Model Updating Results

There were advantages and disadvantages to modal data tuning and direct use of FRFs. Tuning (updating) using FRFs directly required careful selection of tuning points and a damping model.

There were 75 tuning points selected, all of which were kept away from the resonance and anti-resonance peaks. Linear tuning parameters were chosen to modify the stiffness and mass matrices of entire substructures. In the end, the penalty function became a non-linear function of these parameters, and an iterative simplex search routine was used to find the best fit between the measured and predicted FRFs.

After tuning the baseline model, the penalty function showed significant improvement in the correlation between the measured and predicted FRFs. Even though there was good correlation between the FRFs, physical significance was lost in the model from the large changes in the parameters of the model. For example, the batten's mass was increased by a factor of 2.688 in the updating process. Changes like these made the baseline FRF-tuned model questionable because the mass was a measured property.

Tuning the stiff model did not require large deviations from the measured properties of the structure. The time it took to tune the models went from 200 minutes for the baseline FRF-tuned model to 900 minutes on average for the stiff FRF-tuned model. Table 9 shows the value of the of the penalty function measured at the tuning points.

Table 9. Penalty function values before and after tuning

Finite Element Model	Penalty Function Value
Baseline Model	110.3
Baseline Modal-Tuned Model	36.8
Baseline FRF-Tuned	26.6
Stiff Model	29.1
Stiff FRF-Tuned	18.4

6.1.3 Pattern Classifier Results

Various pattern classifiers were used to identify different damage states from the measured data. First, features were extracted from the accelerometer output. These feature vectors compressed the

data 40:1 while retaining the salient features that made the vibrational response from each damage state unique. The following section presents the results from training and testing the pattern classifiers using this new feature set.

The variation in the measured FRFs was studied and assumed to be normally distributed. A normally distributed random number generator using the measured variance with one synthetic FRF was used to create training FRFs. Initially, the pattern classifiers were trained using a set of 1,650 measured feature vectors and tested using a different set of 1,650 measured feature vectors. All five pattern classifiers tested without error which confirmed the selection of the feature extraction method and pattern classifiers for this set of data. The pattern classifiers were then trained using synthetic data from the baseline modal-tuned, baseline FRF-tuned, and stiff FRF-tuned models.

The initial test results showed the MLP did not produce better results than the Gaussian classifier for this type of data set. The MLP was tested because most previous researchers used it, but a simple Gaussian classifier was more than sufficient. Compared to an MLP, the Gaussian classifier was much quicker and easier to use. Table 10 shows the accuracy rate from 3,300 measured test vectors against the four variations of Gaussian classifiers trained using 9,900 synthetic training vectors from each of the three tuned FE models.

Table 10. Accuracy rate from 300 synthetic training vectors from each class and 100 measured test vectors from each class results

FE Model vs Classifier	Baseline Modal-Tuned	Baseline FRF-Tuned	Stiff FRF-Tuned
Grand Diagonal (G-D)	80%	62%	66%
Grand Full (G-F)	78%	66%	79%
By Class Diagonal (BC-D)	82%	64%	77%
By Class Full (BC-F)	73%	66%	74%
Average	78.25%	64.5%	74%

Since it was shown that the classifiers had sufficient features to distinguish each class, the source for the high error rate came from the FE model's inability to model the different damage

states correctly. There were many potential sources of errors in the FE model such as parameters, model order, testing, noise, or measurement. One of the unexpected sources of error came from local modes created by broken members. Unexpected peaks showed up the damaged FRFs that could not initially be accounted for. After careful analysis, it was found that the excited disconnected members on the FTE showed up in the FRFs as a local modes. To model any type of break measurements, each type of break would have to be measured and then these breaks could be lumped together as one damage class.

Initially, this damage detection process used two accelerometers at node 36. One accelerometer was parallel to the shaker and the other orthogonal. The orthogonal accelerometer did not match up well because of the complex damping characteristics of the structure. It was discovered that using the accelerometers directly under the shakers and in the same orientation gave superior results. There was about a 10% decrease in the error rate for all pattern classifiers for all 33 damage states.

Finally, the misclassification errors were carefully examined and broken into three groups: related member, unbroken, and unrelated. Table 11 shows how the error rate broke down for the by class diagonal Gaussian classifier for each of the models. Clearly the baseline modal-tuned model gave the best results because it had the majority of its error rate in the related members category. If the goal was to identify the member or the related member the final accuracy rate would have been 97% for the baseline modal-tuned model, 88% for the baseline FRF-tuned model, and 86% for the stiff FRF-tuned model.

Table 11. Error rate from By Class Diagonal

By Class Diagonal Classifier	Baseline Modal-Tuned	Baseline FRF-Tuned	Stiff FRF-Tuned
Initial error rate	18%	36%	23%
Related member errors	15%	24%	9%
Unbroken errors	2%	11%	10%
Unrelated errors	1%	1%	4%

6.2 Research Conclusions and Recommendations

The research focused on developing and tuning finite element models to train pattern classifiers to detect and locate damage in a real structure. The majority of the research time was spent developing and tuning various FE models. The baseline FRF-tuned model could never produce great results because the large changes in the tuning parameters made the model lose its physical significance and did not do a good job of modelling the different damage states.

Unfortunately, the FE models were not able to produce perfect results. Given more time, the FE models would have been improved by more careful characterization of the joint conditions and damping. To reduce errors in modelling, more time should be spent measuring the characteristics of the joints instead of counting on the updating routines to figure them out. For example, a single joint that included a plate could be cut out of one of the remaining sections of the truss and fully characterized. Since modal data tuning yielded better results in the final damage detection process, this method should be preferred when a good damping model is unattainable. The main disadvantage of the direct use of FRFs for tuning in this research was the inability to model damping correctly. Since the modal data tuning method does not require damping in the tuning process, it works better for the FTE. Additionally, it was noted that the tuning algorithm should be used as a tool to adjust unknown parameters, not the known parameters.

In the pattern classifier training and testing phase, a new feature set was developed and shown to capture the salient properties of each damage state making them uniquely identifiable. It was also shown that a simple Gaussian classifier could easily handle discerning the differences between these states, and the sophisticated capabilities of an MLP was not required.

The key factors that would make this type of damage detection more feasible would be to use it on simpler structures that could be more accurately modelled or structures that could be modified to different states and measured. For example, the different failures of a machine that had ball

bearings could be measured for bearing failure by removing bearings one at a time, and the data could be measured. Future attempts using pattern recognition should take into account the various ways damage can be created. This method shows great promise because the method could easily be made more precise by adding in more accelerometer readings, more accurate modelling techniques, and more powerful classifiers.

APPENDIX A - Flexible Truss Experiment Properties and Calculations

A.1 Flexible Truss Experiment

The basic structure of the experiment is a lightly damped 6-meter truss cantilevered vertically from a rigid base. The truss is assembled from two equal length frames of welded tubular aluminum alloy longerons and battens with bolt-in tubular Lexan diagonals in a back-to-back "K" pattern. The truss has a square cross section 20 inches on a side.

A.2 Material Properties

The longerons and battens of the FTE are made from 6061-T6 aluminum alloy. The diagonal members are made from low-loss Lexan tubing with aluminum end sockets that are bolted at the vertical plates with steel bolts. The following tables list the properties of the key materials in the experiment.

Table 12. Aluminum properties

Aluminum Properties	
Mass density (ρ)	2.699 g/cm ³ or 2.54E-4 lb _m /in ³
Modulus of elasticity (E)	9.9E+6 lb _f /in ²
Modulus of elasticity in shear (G)	3.8E+6 lb _f /in ²

Table 13. Lexan properties

Lexan Properties	
Mass density (ρ)	.99 g/cm ³ or 9.32E-5 lb _m /in ³
Modulus of elasticity (E)	5.01E+5 lb _f /in ²
Modulus of elasticity in shear (G)	1.5E+5 lb _f /in ²

A.3 Longerons

There are 32 longerons in the FTE. The longerons are 6061-T6 aluminum alloy tubes with a 1.5 inch square cross section and a 0.065 inch wall thickness.

$$Area = 1.5^2 - (1.5 - (.065) \cdot 2)^2 = .3731 \text{ in}^2$$

$$I_y = I_x = \frac{1}{12} \text{base} \times \text{height}^3 = \left[\frac{1}{12} (1.5)^4 - \frac{1}{12} (1.5 - (.065) \cdot 2)^4 \right] = .12831 \text{ in}^4$$

$$\begin{aligned} J &= \frac{1}{12} \times 1.5^2 (2 \times 1.5^2) - \frac{1}{12} \times (1.5 - (.065) \cdot 2)^2 (2 \times (1.5 - (.065) \cdot 2)^2) \\ &= 0.25662 \text{ in}^4 \end{aligned}$$

A.4 Regular and Top Battens

There are 24 regular battens and 8 top battens in the FTE. The battens are 6061-T6 tubes with 0.50 inch square cross section and 0.063 inch wall thickness.

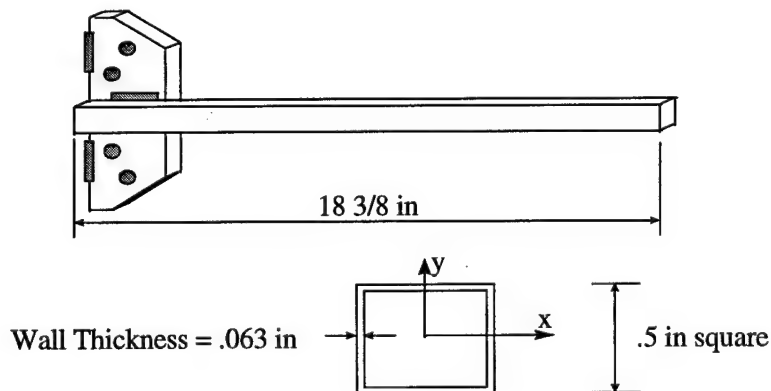


Figure 27. Regular and top batten dimensions

$$\text{area} = \text{height}^2 - (\text{height} - 2 \times \text{thickness})^2 = .5^2 - (.5 - (.063) \cdot 2)^2 = .11012 \text{ in}^2$$

$$I_y = I_x = \frac{1}{12} \text{height}^4 - \frac{1}{12} (\text{height} - 2 \times \text{thickness})^4$$

$$= \left[\frac{1}{12} (.5)^4 - \frac{1}{12} (.5 - (.063) \cdot 2)^4 \right] = 3.5779 \times 10^{-3} \text{ in}^4$$

$$\begin{aligned} J &= \frac{1}{6} \text{height}^4 - \frac{1}{6} (\text{height} - 2 \times \text{thickness})^4 \\ &= \frac{1}{6} \times .5^4 - \frac{1}{6} (.5 - (.063) \cdot 2)^4 = 7.1558 \times 10^{-3} \text{ in}^4 \end{aligned}$$

A.5 Mid Battens

There are 4 mid battens in the FTE. They are similar to the regular battens except they are together in pairs resulting in much larger I_x and I_z values.

$$\begin{aligned} \text{area} &= \left[\text{height}^2 - (\text{height} - 2 \times \text{thickness})^2 \right] \times 2 \\ &= \left(.5^2 - (.5 - (.063) \cdot 2)^2 \right) \times 2 = .22025 \text{ in}^2 \\ I_y &= \frac{1}{12} \text{base} \times \text{height}^3 = \left[\frac{1}{12} (.5)^4 - \frac{1}{12} (.5 - (.063) \cdot 2)^4 \right] \times 2 = 7.1558 \times 10^{-3} \text{ in}^4 \end{aligned}$$

$$\begin{aligned} I_x &= \frac{1}{12} \text{base} \times \text{height}^3 + \text{dist}^2 \times \text{area} \\ &= \left[\left[\frac{1}{12} (.5)^4 - \frac{1}{12} (.5 - (.063) \cdot 2)^4 \right] + (.5^2 \times .22025) \right] \times 2 = 0.11728 \text{ in}^4 \end{aligned}$$

$$\begin{aligned} J &= \left[\frac{1}{6} \text{height}^4 - \frac{1}{6} (\text{height} - 2 \times \text{thickness})^4 + (\text{distance to axis})^2 \times \text{area} \right] \times 2 \\ &= \left[\frac{1}{6} \times .5^4 - \frac{1}{6} (.5 - (.063) \cdot 2)^4 + (.5^2 \times .22025) \right] \times 2 = .12444 \text{ in}^4 \end{aligned}$$

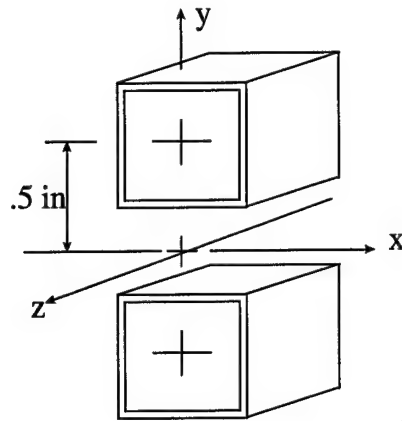


Figure 28. Mid-batten dimensions

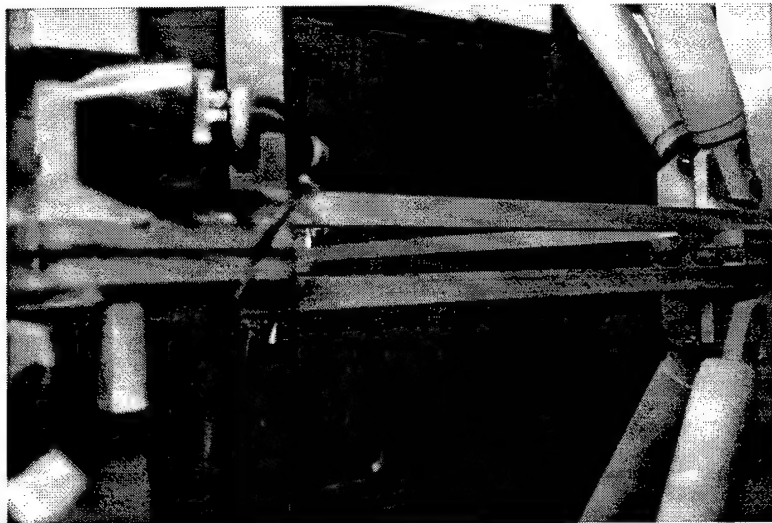


Figure 29. Picture of mid-batten

A.6 Diagonal Members

There are 32 diagonal members in the FTE. They are made from Lexan tubing with a circular cross section on 1.5 inch diameter and have a 0.125 inch wall thickness. The diagonals have aluminum end fittings and are fastened to the truss frame with two steel bolts and a half-clevis joint at

each end.

$$\begin{aligned} \text{area} &= \pi \times (\text{radius}^2 - (\text{radius} - \text{thickness})^2) = \pi \times (.75^2 - (.75 - .125)^2) = .53996 \text{ in}^2 \\ I_x &= I_y = \frac{1}{4} \cdot \pi \cdot (\text{radius}^4 - (\text{radius} - \text{thickness})^4) = \frac{1}{4} \cdot \pi \cdot \left(\left(\frac{1.5}{2} \right)^4 - \left(\frac{1.5}{2} - .125 \right)^4 \right) \\ &= .12866 \text{ in}^4 \end{aligned}$$

$$J = 2 \times I_x = .25732 \text{ in}^4$$

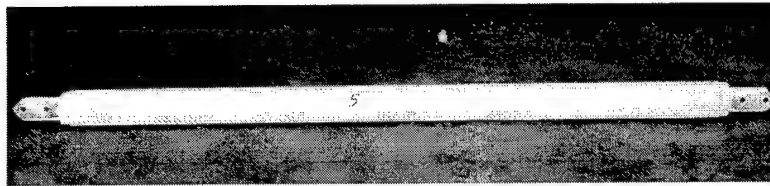


Figure 30. Picture of diagonal member

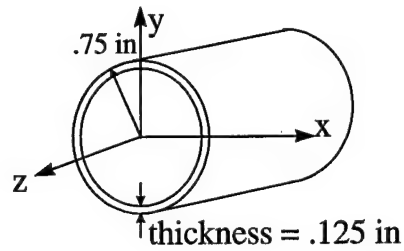


Figure 31. Diagram of diagonal cross section

A.7 Vertical Plates

There are 32 vertical plates welded to the longerons and battens used to connect the diagonals to the truss.

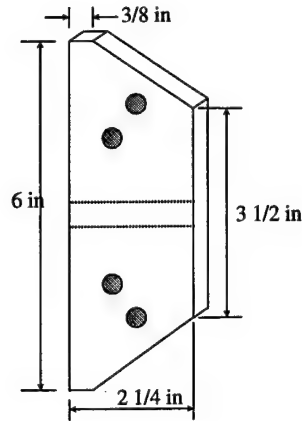


Figure 32. Diagram of vertical plate

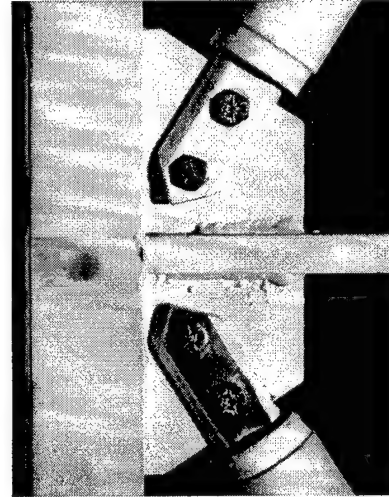


Figure 33. Picture of truss edge

$$\text{area} = \text{height} \times \text{width} - 2 \times \text{triangular area}$$

$$= (6 \times 2.25) - 2 \times \left(\frac{1}{2} \times \left(\frac{6 - 3.5}{2} \right) \times \left(2.25 - \frac{3}{8} \right) \right) = 11.156 \text{ in}^2$$

$$\text{volume} = \text{area} \times \text{thickness} - 4 \text{ holes} = 11.156 \times \frac{5}{16} - 4 \times \pi \times \left(\frac{1}{4} \right)^2 \times .25 = 3.2899 \text{ in}^3$$

$$\text{mass} = \text{volume} \times \text{density} = 3.2899 \text{ in}^3 \times 2.54 \times 10^{-4} \text{ lb}_m/\text{in}^3 = 8.3563 \times 10^{-4} \text{ lb}_m$$

A.8 Horizontal Plates

There are 8 horizontal plates at the middle of the truss and 4 at the top and four on the bottom. The horizontal plates are used to bolt the section of the truss together and attach the shakers on the top.

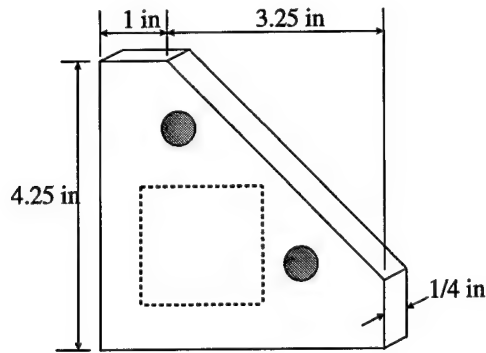


Figure 34. Diagram of top plate

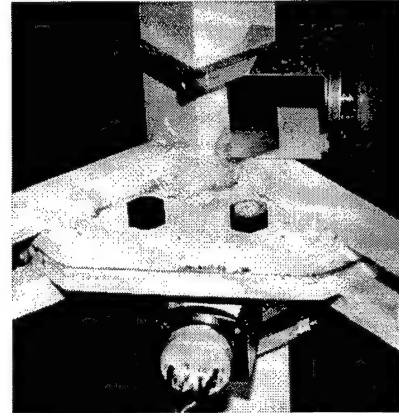


Figure 35. Picture of mid plate

$$\text{area} = \text{height} \times \text{width} - \text{triangular area} = 4.25^2 - \frac{1}{2} \cdot 3.25^2 = 12.781 \text{ in}^2$$

$$\text{volume} = \text{area} \times \text{thickness} - 2 \text{ holes} = 12.781 \times .25 - 2 \times \pi \times \left(\frac{3}{8}\right)^2 \times .25 = 2.9744 \text{ in}^3$$

$$\text{mass} = 2.9744 \text{ in}^3 \times 2.54 \times 10^{-4} \frac{\text{lb}_m}{\text{in}^3} = 7.555 \times 10^{-4} \text{ lb}_m$$

A.9 Shakers

The shakers are attached to the top of the structure and are orthogonal to each other to excite as many modes as possible. The shakers use a linear dc motor with the armature fixed to the base. There are dual shafts attached to the mass to keep it aligned, with a 1 inch travel in either direction. The mass is centered by light springs that do create a 2 Hz resonance but have negligible effect after 4Hz.

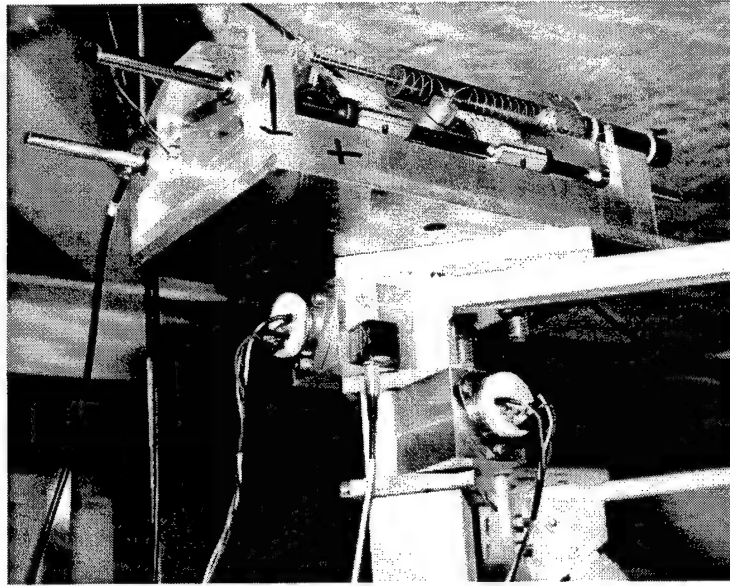


Figure 36. Picture of shaker and accelerometers

$$I_x = 0.0748 \text{ in}^4$$

$$I_y = 0.2581 \text{ in}^4$$

$$J = 0.2406 \text{ in}^4$$

$$\text{Mass} = 2.7 \times 10^{-2} \text{ lb}_m$$

A.10 Additional Lumped Masses

The following items added additional mass to the structure and were added to the FE models as lumped masses to the closest nodes.

$$\text{Big Bolts Mass} = 33.8 \text{ grams} * 5.742E - 6 \text{ lb}_m \times \text{grams}^{-1} = 1.94 \times 10^{-4} \text{ lb}_m$$

$$\text{Small Bolts Mass} = 13.5 \text{ grams} * 5.742E - 6 \text{ lb}_m \times \text{grams}^{-1} = 7.75 \times 10^{-5} \text{ lb}_m$$

$$\text{Accelerometers} = 176.1 \text{ grams} * 5.742E - 6 \text{ lb}_m \times \text{grams}^{-1} = 1.0112 \times 10^{-3} \text{ lb}_m$$

$$\text{Aluminum End Sockets} = 204.6 \text{ grams} * 5.742E - 6 \text{ lb}_m \times \text{grams}^{-1} = 1.1748 \times 10^{-3} \text{ lb}_m$$

$$\text{Aluminum Welding per inch} = .25^2 \times .5 \times 2.54 \times 10^{-4} \text{ lb}_m \times \text{in}^{-3} = 7.9375 \times 10^{-6} \text{ lb}_m \times \text{in}^{-1}$$

A.11 Boundary Conditions

The boundary conditions for the FTE were to restrict all degrees of freedom on the first four nodes of the model. As seen in Figure 37, the base of the FTE is securely attached to a large aluminum plate which is bolted to the concrete floor.

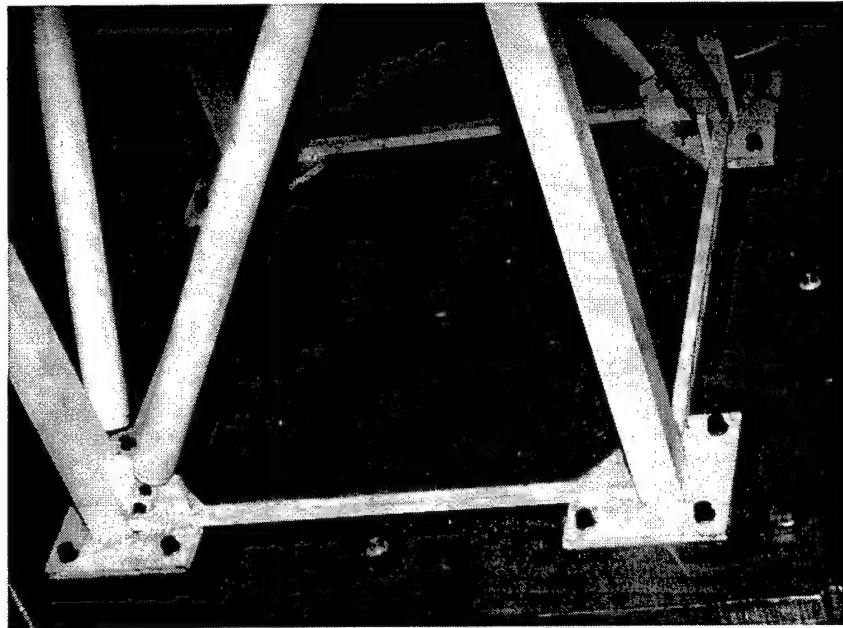


Figure 37. Picture of the FTE Base

APPENDIX B - Element Matrices Calculations

The following is a development of the element mass and stiffness matrices for a horizontal element. The matrices are transformed by rotation matrices to match the elements orientation in three dimensional space before assembling the global mass and stiffness matrices. This mathematical development was included to assist the reader in better understanding the finite element model construction and tuning.

B.1 Stiffness Matrix

The matrix $K_{element}$ is constructed from the following matrices,

$$\begin{aligned}
 K_1 &= \begin{bmatrix} \frac{AE}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{12EI_1}{L^3} & 0 & 0 & 0 & \frac{6EI_1}{L^2} \\ 0 & 0 & \frac{12EI_2}{L^3} & 0 & \frac{-6EI_2}{L^2} & 0 \\ 0 & 0 & 0 & \frac{I_3G}{L} & 0 & 0 \\ 0 & 0 & \frac{-6EI_2}{L^2} & 0 & \frac{4EI_2}{L} & 0 \\ 0 & \frac{6EI_1}{L^2} & 0 & 0 & 0 & \frac{4EI_1}{L} \end{bmatrix} \\
 K_2 &= \begin{bmatrix} \frac{AE}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{12EI_1}{L^3} & 0 & 0 & 0 & \frac{-6EI_1}{L^2} \\ 0 & 0 & \frac{12EI_2}{L^3} & 0 & \frac{6EI_2}{L^2} & 0 \\ 0 & 0 & 0 & \frac{I_3G}{L} & 0 & 0 \\ 0 & 0 & \frac{6EI_2}{L^2} & 0 & \frac{2EI_2}{L} & 0 \\ 0 & \frac{-6EI_1}{L^2} & 0 & 0 & 0 & \frac{2EI_1}{L} \end{bmatrix} \\
 K_3 &= \begin{bmatrix} \frac{AE}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{12EI_1}{L^3} & 0 & 0 & 0 & \frac{6EI_1}{L^2} \\ 0 & 0 & \frac{12EI_2}{L^3} & 0 & \frac{-6EI_2}{L^2} & 0 \\ 0 & 0 & 0 & \frac{I_3G}{L} & 0 & 0 \\ 0 & 0 & \frac{-6EI_2}{L^2} & 0 & \frac{2EI_2}{L} & 0 \\ 0 & \frac{6EI_1}{L^2} & 0 & 0 & 0 & \frac{2EI_1}{L} \end{bmatrix} \\
 K_4 &= \begin{bmatrix} \frac{AE}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{12EI_1}{L^3} & 0 & 0 & 0 & \frac{-6EI_1}{L^2} \\ 0 & 0 & \frac{12EI_2}{L^3} & 0 & \frac{6EI_2}{L^2} & 0 \\ 0 & 0 & 0 & \frac{I_3G}{L} & 0 & 0 \\ 0 & 0 & \frac{6EI_2}{L^2} & 0 & \frac{4EI_2}{L} & 0 \\ 0 & \frac{-6EI_1}{L^2} & 0 & 0 & 0 & \frac{4EI_1}{L} \end{bmatrix}
 \end{aligned}$$

are assembled as,

$$K_{element} = \begin{bmatrix} K_1 & K_2 \\ K_3 & K_4 \end{bmatrix}$$

B.2 Mass Matrix

The element \mathbf{M} matrix was constructed from the shape functions for the beam element used in the Structural Dynamics Toolbox. The following is the pieces of the consistent mass matrices,

$$\begin{aligned} \mathbf{M}_1 &= \frac{\rho AL}{420} \begin{bmatrix} 140 & 0 & 0 & 0 & 0 & 0 \\ 0 & 156 & 0 & 0 & 0 & 22L \\ 0 & 0 & 156 & 0 & -22L & 0 \\ 0 & 0 & 0 & 0.00007 & 0 & 0 \\ 0 & 0 & -22L & 0 & 4L^2 & 0 \\ 0 & 22L & 0 & 0 & 0 & 4L^2 \end{bmatrix} \\ \mathbf{M}_2 &= \frac{\rho AL}{420} \begin{bmatrix} 70 & 0 & 0 & 0 & 0 & 0 \\ 0 & 54 & 0 & 0 & 0 & -130l \\ 0 & 0 & 54 & 0 & 130l & 0 \\ 0 & 0 & 0 & 0.00035 & 0 & 0 \\ 0 & 0 & -130l & 0 & -3l^2 & 0 \\ 0 & 130l & 0 & 0 & 0 & -3l^2 \end{bmatrix} \\ \mathbf{M}_3 &= \frac{\rho AL}{420} \begin{bmatrix} 70 & 0 & 0 & 0 & 0 & 0 \\ 0 & 54 & 0 & 0 & 0 & 130L \\ 0 & 0 & 54 & 0 & -130L & 0 \\ 0 & 0 & 0 & 0.00035 & 0 & 0 \\ 0 & 0 & 130L & 0 & -3L^2 & 0 \\ 0 & -130L & 0 & 0 & 0 & -3L^2 \end{bmatrix} \\ \mathbf{M}_4 &= \frac{\rho AL}{420} \begin{bmatrix} 140 & 0 & 0 & 0 & 0 & 0 \\ 0 & 156 & 0 & 0 & 0 & -22L \\ 0 & 0 & 156 & 0 & 22L & 0 \\ 0 & 0 & 0 & 0.00007 & 0 & 0 \\ 0 & 0 & 22L & 0 & 4L^2 & 0 \\ 0 & -22L & 0 & 0 & 0 & 4L^2 \end{bmatrix} \end{aligned}$$

and they are assembled as,

$$\mathbf{M}_{element} = \begin{bmatrix} \mathbf{M}_1 & \mathbf{M}_2 \\ \mathbf{M}_3 & \mathbf{M}_4 \end{bmatrix}$$

to create the final consistent mass matrix for the element.

Bibliography

- [1] E. Balmes. A finite element updating procedure using frequency response functions. applications to the mit/serc interferometer testbed. *Proceedings of the 11th International Modal Analysis Conference*, pages 176–182, 1993.
- [2] E. Balmes. *Structural Dynamics Toolbox User Manual*. Scientific Software Group, 1997.
- [3] S. Barai and P. Pandey. Performance of the generalized delta rule in structural damage detection. *Engineering Applications of Artificial Intelligence*, 8(2):211–221, 1995.
- [4] R. Cobb. *Structural Damage Identification From Limited Measurement Data*. PhD thesis, Air Force Institute of Technology, 1996.
- [5] S. Doebling, C. Farrar, M. Prime, and D. Shevitz. Damage identification and health monitoring of structural and mechanical system from changes in their vibration characteristics: A literature review. 1996.
- [6] R. Duda and P. Hart. *Pattern Classification and Scene Analysis*. John Wiley and Sons, 1973.
- [7] M. Elkordy, K. Chang, and G. Lee. Neural network trained by analytically simulated damage states. *ASCE Journal of Computing in Civil Engineering*, 7(2):130–145, 1993.
- [8] M. I. Friswell and J. E. Mottershead. *Finite Element Model Updating and Structural Dynamics*. Kluwer: Academic Press, 1995.
- [9] R. Gordon. The 12-meter truss active control experiment design, analysis, and open-loop testing. Technical Report WL-TR-92-3012, Flight Dynamics Directorate, Wright Laboratory, Air Force Systems Command, Wright-Patterson AFB, OH, 1992.
- [10] P. Kirkegaard and A. Rytter. Use of neural networks for damage assessment in a steel mast. *Proceedings of the 12th International Modal Analysis Conference*, pages 1128–1134, 1994.
- [11] S. E. Klenke and T. L. Paez. Damage identification with probabilistic neural networks. *Proceedings of the 14th International Modal Analysis Conference*, pages 99–104, 1996.
- [12] S. Lammens, W. Heylen, and P. Sas. The selection of updating frequencies and the choice of a damping approach for model updating procedures using frequency response functions. *Proceedings of the 12th International Modal Analysis Conference*, pages 1383–1393, 1994.
- [13] J. Lifshitz and A. Rotem. Determination of reinforcement unbonding of composites by vibration technique. *Journal of Composite Materials*, pages 412–423, 1969.
- [14] R. Manning. Damage detection in adaptive structures using neural networks. *Proceedings of the 35th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, pages 160–172, 1994.
- [15] J. Mottershead and M. Friswell. Model updating in structural dynamics: A survey. *Journal of Sound and Vibration*, 167(2):347–375, 1993.
- [16] H. Natke and C. Cempel. *Model-Aided Diagnosis of Mechanical Systems*. Springer Verlag, Berlin, GE, 1997.
- [17] J. Nelder and R. Meade. A simplex method for function minimization. *Computer Journal*, 7:308–313.

- [18] C. Povich and T. Lim. An artificial neural network approach to structural damage detection using frequency response functions. *Proceedings of the 35th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, pages 151–159, 1994.
- [19] L. Rabiner and B. Juang. *Fundamentals of Speech Recognition*. Prentice-Hall Inc., 1993.
- [20] Scientific-Atlanta, Inc. *Pro Series Dynamic Signal Analyzer SA390 Operators Manual*, dcn 222623 revision a edition, March 1996.
- [21] J. Stephens and R. VanLuchene. Integrated assessment of seismic damage in structures. *Microcomputers in Civil Engineering*, 9:119–128, 1994.
- [22] P. Szewczyk and P. Hajela. Damage detection in structures based on feature-sensitive neural networks. *ASCE Journal of Computing in Civil Engineering*, 8(2):163–178, 1994.
- [23] X. Wu, J. Ghaboussi, and J. Garrett. Use of neural networks in detection of structural damage. *Computers and Structures*, 42(4):649–659, 1992.

Vita

Eric D. Swenson was born on April 11, 1967 in Iowa City, Iowa. He enlisted in the United States Air Force in August 1985. After completing a year long Systems Repair Technician technical school at Lowry AFB, CO. Airman Swenson went to McClellan Central Labs at McClellan AFB, CA to work as an electronics technician. Three years later, Sergeant Swenson was selected for the Airmen Educational Commissioning Program. He went to Ohio State University in Columbus, OH where he graduated Magna Cum Laude with Honors in Civil Engineering in March 1993. Officer Trainee Swenson then went to Officer Training School at Lackland AFB where he graduated as a distinguished graduate in July 1993. After commissioning, Second Lieutenant Swenson went to Holloman AFB, NM where he served as a Civil Engineering Officer in the 49th Civil Engineer Squadron. His responsibilities included Deputy Flight Chief of the Environmental Engineering Flight, Project Engineer in the Engineering Flight, and Flight Chief of the Resources Flight. First Lieutenant Swenson was selected to attend the Air Force Institute of Technology Graduate School of Engineering to earn a Masters of Science in Astronautical Engineering Degree. The completion of his degree will prepare him for his future job as a Project Engineer at Cape Canaveral Air Force Station, FL. Captain Eric Swenson married his high school sweetheart Kristine A. Leslie in 1985 and has two sons, Sean and Seth.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Please reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE March 1998	3. REPORT TYPE AND DATES COVERED Masters Thesis	
4. TITLE AND SUBTITLE Damage Detection Using Pattern Classifiers			5. FUNDING NUMBERS	
6. AUTHOR(S) Eric D. Swenson, Capt, USAF				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology, 2950 P Street WPAFB OH 45433-77			8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/GA /ENY/98M-01	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Captain Richard Cobb Phone(505)846-1340 AFRL/VSDV 3550 Aberdeen Ave SE Kirtland AFB, NM 87117-5776			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES Major Jeffrey Turcotte/(937)255-6565/jturcott@afit.af.mil				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited			12b. DISTRIBUTION CODE A	
13. ABSTRACT (Maximum 200 words) The research focused on developing and tuning finite element (FE) models to train pattern classifiers to detect and locate damage in a real structure. The method was tested on the Flexible Truss Experiment at the Air Force Institute of Technology. A low order FE model called the baseline model and a high order model called the stiff model were created and tuned using measured Frequency Response Functions (FRFs) in the Structural Dynamics Toolbox for MATLAB. Features (or properties) of the FRFs were extracted through an adapted feature extraction process commonly used in speech processing. This new feature set was developed to reduce the amount of data by a factor of 40 while retaining the salient properties that made the changes in the FRFs unique to each damage state. To prove that the different features extracted from 32 damage states were unique, some initial tests were performed in which five classifiers were trained and tested using measured data. These tests resulted in no classification errors. Using the FE model generated FRFs, the Gaussian classifiers had an average accuracy rate of 78.25% for locating the correct damaged member and 97% accuracy for locating the correct region of damage.				
14. SUBJECT TERMS Damage Detection, Pattern Classifiers, Feature Extraction, Vibrations, Structural Dynamics, Finite Elements,			15. NUMBER OF PAGES 96	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	